Holistic Interpretation of Visual Data based on Topology

Semantic Segmentation of Architectural Facades

Radwa Fathalla

A thesis presented for the degree of

Doctor of Philosophy

Aston University

June 2017

Thesis Summary

Holistic Interpretation of Visual Data based on Topology
Semantic Segmentation of Architectural Facades
Radwa Fathalla
Doctor of Philosophy
June 2017

The work presented in this dissertation is a step towards effectively incorporating contextual knowledge in the task of semantic segmentation. To date, the use of context has been confined to the genre of the scene with a few exceptions in the field. Research has been directed towards enhancing appearance descriptors. While this is unarguably important, recent studies show that computer vision has reached a near-human level of performance in relying on these descriptors when objects have stable distinctive surface properties and in proper imaging conditions. When these conditions are not met, humans exploit their knowledge about the intrinsic geometric layout of the scene to make local decisions. Computer vision lags behind when it comes to this asset. For this reason, we aim to bridge the gap by presenting algorithms for semantic segmentation of building facades making use of scene topological aspects.

We provide a classification scheme to carry out segmentation and recognition simultaneously. The algorithm is able to solve a single optimization function and yield a semantic interpretation of facades, relying on the modeling power of probabilistic graphs and efficient discrete combinatorial optimization tools.

We tackle the same problem of semantic facade segmentation with the neural network approach. We attain accuracy figures that are on-par with the state-of-the-art in a fully automated pipeline. Starting from pixelwise classifications obtained via Convolutional Neural Networks (CNN). These are then structurally validated through a cascade of Restricted Boltzmann Machines (RBM) and Multi-Layer Perceptron (MLP) that regenerates the most likely layout.

In the domain of architectural modeling, there is geometric multi-model fitting. We introduce a novel guided sampling algorithm based on Minimum Spanning Trees (MST), which surpasses other propagation techniques in terms of robustness to noise. We make a number of additional contributions such as measure of model deviation which captures variations among fitted models.

Index terms— Geometric model fitting, Deep learning, Meta-learning, Layout, Contextual image partitioning

Dedication

To my sons, Ismail and Aly.

Acknowledgements

First of all, deepest gratitude is due to Dr. George Vogiatzis for his continuous guidance throughout the course of this work and his generously offered advice on how to conduct research. Also, I would like to express my appreciation to the valuable comments of Prof. Ian Nabney.

Besides my advisor, I would like to thank my examining committee at the viva Dr. Henrik Aanæs and Dr. Yulan He, for their insightful comments and questions which motivated me to improve my analysis and resulted in a better version of this thesis.

I am grateful for the Arab Academy for Science and Technology for financially supporting my studies. I am especially thankful to my professors: Prof. Yasser El Sonbaty for giving me the opportunity to join this program and his sustained support, Prof. Ossama Badawy for his moral encouragement that kept my work momentum, Prof. Moustafa Hussien for understanding research needs and backing us up to have an adequate research environment. Also, many thanks go to my colleagues and staff at the Arab Academy for being such a good company.

Above all, I am truly indebted to my father and mother for their lifelong care and sacrifices. Also, I am profoundly grateful to my husband for his understanding and compassion. Thanks go to my sister and her family for the joy they bring to my life. Last but not least, I thank my mother-in-law for her efforts to spare me the time for study.

Finally, my love goes to my grandfather, my role model and my lifelong source of inspiration.

Contents

1 Introduction **12**

1.1 Aim of the work 12

1.2 Background 13

1.3 Rationale of the work 18

1.4 Contributions and structure of the thesis 21

2 Generic Semantic Segmentation **23**

2.1 Overview 23

2.2 Scene Parsing 24

2.3 Object Localization 28

2.4 CNNs 33

3 Facade Semantic Segmentation **39**

3.1 Feature vector classification 39

3.2 Expert-based layout enhancement 41

3.3 Parsing grammars 41

3.4 Repetitive patterns 44

3.5 Regularized optimization functions 48

4 Probabilistic Graphical Models and Semantic Segmentation **53**

4.1 Random Field Optimization 54

4.1.1 Graphcut Algorithm 55
4.1.2 Sequential Tree ReWeighted (TRW-S) message passing 58
4.2 Restricted Boltzmann Machine 59

5 Geometric Multi-model fitting 65
5.1 Introduction 65
5.2 A brief review: existing approaches 65
5.2.1 Energy formulation 66
5.2.2 Similarity-formulation 71

6 Multi-model fitting based on Minimum Spanning Tree 73
6.1 Introduction 73
6.2 Proposed Algorithm 74
6.2.1 Minimum Spanning Tree (MST) guided sampling 75
6.2.2 Multiplicity guided model detection 82
6.3 Experimental Evaluation 85
6.3.1 MST guided sampling 87
6.3.2 Multiplicity guided model detection 87
6.4 Conclusion 88

7 Optimization of Facade Segmentation Based on Layout Priors 90
7.1 Introduction 90
7.2 Facade Segmentation Optimization 92
7.2.1 Appearance Cues 95
7.2.2 Layout Cues 96
7.2.3 Learning the Hyperparameters. 102
7.3 Evaluation 104
7.4 Conclusion 107

8 A Deep Learning Pipeline for Semantic Facade Segmentation 109

8.1 Introduction 109

8.2 Proposed Algorithm 111

8.2.1 Appearance cues 112

8.2.2 Structure cues 113

8.3 Evaluation 121

8.4 Conclusion 130

9 Discussion and Conclusion 131

9.1 Summary 132

9.2 Future extensions 134

List of Figures

3.1 A sample of 2 images illustrating the difference between (a) a proper lattice and (b) repeated structures. 45

5.1 Categorization of work done in model fitting 66

5.2 A snapshot of point arrangement showing 3 randomly formed models (lines). The points in blue share very similar preference based on the cross structure despite belonging in different models. 71

6.1 A snapshot of point arrangement showing 2 inliers in green with an in-between distance larger than the distances between one of them and the gross outliers in red. . . 74

6.2 (a) Graphs showing the index of the point added at each iteration of the expansion of the sample set based on Front propagation; (b) based on Minimum spanning tree. In this example, points with indices > 6000 are noise points. It is evident that the bulk of noise points are accessed at late iterations after all the models points have been visited. 77

6.3 (a) MST of size z initiated at some point. (b) Subset of the MST that satisfies our criteria. 77

6.4 MS sub-T selection criteria (a) A model deviation signal showing a typical behavior of models constructed from subsets of the sample set; (b) A graph showing the smoothed model deviation, margin of error, ground truth residual. Two vertical lines marking the valley of interest. The circle shape marks the chosen subtree size for this point. . 79

6.5 Normalized d_{0i} between the rotating line and the x-axis at each angle position. 80

6.6 Wadham. (a) Ground truth (outlier points added shown as red circles). Results of (b) proposed algorithm; (c) J-linkage; (d) PEARL (results are different from those reported in their paper because utilized parameters of energy function were not given). 86

6.7 Merton college III. (a) Ground truth (outlier points added shown as yellow crosses). Results of (b) proposed algorithm; (c) J-linkage; (d) PEARL. 88

6.8 Dino books. Result of (a) proposed algorithm; (b) post processing of outlier residual filtering increases precision to 0.7001; (c) J-linkage; (d) PEARL. 88

7.1 Diagram showing proposed system modules and their interactions. 93

7.2 A sample of 2*d* adjacency histogram in the vertical direction. It indicates that the most adundant type of adjacency in this image patch is a facade underneath a window structure (indicated by the bright red color). 98

7.3 (a) A sample of a watershed segmented image that reveals how small the superpixels are. (b) A sample of long-range edges shown in red. (c) A sample of short-range edge approximated cost matrix for the CMP dataset [1]. Structure 1 incurs the least cost, which signals that it is the most frequently encountered structure in this image patch. The most abundant transition is between structures 7 and 2. Structures 4, 6, 8, and 11 are never seen in this image patch during training. Values on the diagonal are in a lower range than the ones on the lower and upper triangles to promote same labeling. 99

7.4 (a) Semi-log scale plot of the cost against PSO iterations. (b) Accuracy plots for the images in ECP dataset when different options for IASC are activated. 101

7.5 Sample outcomes in tabular format. Rows from (1) to (2) ECP-Monge samples; Rows from (3) to (9) CMP samples. Column (a) Ground truth; results of (b)NC; (c) PA; (d) SEP; (e) LEP. 108

8.1 A schematic showing system modules 112

8.2 Final labeling of a sample facade image (a) without dataset augmentation, (b) with dataset augmentation. (c) The groundtruth map. It is clear that the door and chimney structures were correctly recovered in (b) 117

8.3 (a) A sample image. (b) First possible ground truth map. (c) Second possible ground truth map. 123

8.4 A sample graph for the accuracy and objective figures over the training epochs of the CNN. Accuracy is calculated per-pixel against ground truth. The objective is a cost function calculated as the cross entropy between the ground truth and the predicted distribution. 124

8.5 A sample graph for the reconstruction error on the validation set over the training epochs. R_{Φ}^{y} consistently has a higher error than R_{Φ}^{x}, due to the higher variability of vertical scanlines over horizontal ones. 125

8.6 (a) A sample image (left) and the result of DPF-Ψ (right). The output of one of the rounded windows shows that the algorithm can handle *to some extent* the case of architecural structures with rounded boundaries without enforcing right angles and straight lines. (b) A sample of groundtruth (left), DPF-Φ (middle) and DPF-Ψ (right) results on the image. It shows that training on the image was able to recover the missing window because the correct label was propagated to it from similar scanlines. 127

8.7 (a) A sample image. (b) Ground truth map. (c) CNN output I^0. (d) Variant 1. (e) Variant 2. (f) DPF-Ψ. 128

8.8 (a) A sample image. (b) Ground truth map. (c) CNN output I^0. (d) Variant 1. (e) Variant 2. (f) DPF-Ψ. 129

List of Tables

6.1 Average per-model residual of closest group of ground truth model points to their best fitting models in the initial set. 85

6.2 Average recall, precision values and the count of detected models for results of (a) J-linkage, (b) PEARL, (c) Our proposed algorithm 86

7.1 Comparative table of the state-of-the-art bottom-up approaches in facade parsing. Legend: ∘ indicates that layout prior is validated in a heuristic postprocessing step or via genetic-based algorithms, in contrast to • which indicates principled non-heuristic involvement of the prior. ♢ states that the prior is quantified numerically and complements a feature vector that is fed into a classifier. 92

7.2 Average accuracies on datasets. NC: No context (appearance only), AP: Aligned Pairs, APRT: Aligned Pairs Regular Triplets, SH: Structural Heuristics, PA: POTTS Adjacency, ST3: Auto-Context classified, PW3: POTTS Smoothed Auto-Context, SEP: Short-range Edges Prior, and LEP: Layout Edges Prior (short- and long- range). 105

8.1 Standard deviation of accuracies of approaches presented in chapters 7 and 8 for comparison purposes. 124

8.2 Overall pixel accuracies based on appearance cues A_1 and when combined with layout cues A_2. The references marked with $*$ are included for completeness of results and are not suitable for direct comparison. 125

8.3 Per-class pixel accuracy, A_3 (Average class pixel accuracy), and IoU results on the *ECP-Monge* dataset. N/C stands for Not Considered. 126

8.4 Per-class pixel accuracy, A_3 (Average class pixel accuracy), and IoU results on the *CMP* dataset. 127

Chapter 1

Introduction

1.1 Aim of the work

This dissertation is directed towards model fitting in vision related tasks, with a focus on a scene interpretation application. Scene parsing (interpretation) is the categorical subdivision of images into semantically meaningful regions. We primarily focus on the application of parsing of building facades. Facades are particularly appealing for parsing techniques. They exhibit a balance between versatility of design and being governed by an intrinsic layout. Facades are full of scene regularities allowing the pruning of the search space of models, and are challenging enough for evaluating state-of-the-art methods for structural modeling. Facades come in different architectural styles. This leads to variability in size, contained structures, geometric layout and surface textures. However, there are regularities such as adjacency relations, the relative positioning of the architectural components, and existence of repetitive patterns. These regularities impose constraints on the likely labels for the image pixels. Furthermore, architectural regularities correspond to long range dependencies between groupings of pixels in the spatial domain. Therefore architectural facades present a very good test case for studying spatial inference under contextual knowledge, one of the key aims of Computer Vision. In this work, we provide algorithms that aid in the segmentation of building facades in the 2d space. The procedure imparts semantics on the components of the facades and develop a step towards

establishing fully recognized models of buildings.

1.2 Background

Interpretation of visual data is a long standing problem in computer research. The motivation behind the numerous proposed approaches has always been the huge impact of developing automated systems, coupled with the incapacity of the existing methodologies to reach a near-human level of proficiency. The applications are vast; remote sensing, medical imaging, face detection, 3D modeling, surveillance cameras, and robot navigation. Humans carry out vision related tasks with seamless effort, speed, precision and an ability to generalize to stimulus never encountered before. The encouraging news for the computer vision community is there have been recent advances on related frontiers that will definitely lead to a leap in the performance of developed algorithms. On the imaging hardware level, we can find high-tech digital cameras, Li DAR, and Kinect. On another front, the evolution of statistical reasoning in the field of machine learning has been a back bone in dealing with vision challenges. In addition, the wide availability of large scale datasets of images has permitted extensive learning which leads to building of more robust visual models and the creation of benchmarks allowing the comparison of related research and the subsequent setting up of a road map for future efforts.

In recent studies [2], it has been shown that context is an important aspect in disambiguating appearance. Blurred snapshots of objects were given to humans and subjected to a machine learning recognizer. Surprisingly, the recognizer outperformed humans in the setting of narrow scope of the object. When a wider scope of the images was considered, the human superiority was evident in learning and applying higher order potentials. It suggests that state-of-the-art algorithms are very successful in modeling appearance based on color and photometric properties, but are lagging behind in modeling context. So far context has been confined to determining *what* objects are included. There is another fundamental unanswered question, *where* to localize these objects in terms of position and extent depending on scene topology.

One of the earliest attempts to formalize the notion of wholesome approach to scene interpretation is *Gestalt theory*. It states that the human eye sees objects in their entirety before perceiving their individual parts, promoting the notion "The whole is other than the sum of the parts". The gestalt effect is the form-generating capability of our senses, particularly with respect to the visual recognition of figures and whole forms in a noisy world, before perceiving their individual elements. Traditionally, the theory has been viewed as a whole-to-part inference framework. However, recent versions of the theory stipulate that properties of a system can be traced from those of its elements, while element properties depend on their interrelations induced by the system configuration [3]. It is this modern outlook that laid the basis for the discrimination between the holistic and global approach to inference. In [3], the authors explain, the holistic approach is more tailored to inferring information about a complex entity starting from its constituents and the global approach is suitable for an entity of sub-parts. Computationally, the distinction is associated with the processing in a bottom-up direction or a top-down one, respectively. The theory has been regarded as descriptive and not been widely used in the computer vision field due to its lack of a computational perspective. Another theory about disambiguation by concurrently studying the parts and their interactions is *Pattern theory* pioneered by Mumford[4]. It has been suggested by many psychologists that the crude world signals acquired through the human senses are perceived only after being configured through signals from memory and logic. Consequently, the feed-forward bottom-up approach which starts from low-level cues (sensory data) to the high-level cues (brain inferences), is not adequate in modeling the visual perception process. Instead, a feed-back top-down cognitive approach to analyze and synthesize representations of the world entities by comparing the observed signal with the tentative interpretations, is recommended. The processing order is not the only aspect by which Pattern Theory affects an automated visual system design. At the heart of this theory, lies a bundle of probabilistic models for different aspects of perceptual phenomena, that are solved by Bayesian inference. It models image primitives and a set of world deformations, namely: noise and blur, multi-scale superposition, domain warping, and interruptions, which should be taken into consideration.

The work presented in contextual reasoning is directed towards two problems: object detection

and scene parsing. For this reason, we include a review of them in the thesis. The objects to be detected are usually made of sub-parts. Context in this respect is provided for the sub-parts in modeling their the inter-relations . This will lead to better localization and ultimately result in more accurate object detection. The most significant work in this field is Deformable Parts Model (DPM) [5]. The technique is usually applied to animal, face, and car detection. Theoretically, modeling the interactions statistically is appealing to our application. In this way, the architectural structures will correspond to the sub-parts. One quickly realizes there are several challenges. In typical DPM, the instances count of a sub-part type is static and pre-specified in the model. A face has 2 eyes, a pedestrian has one head, a car has 4 wheels and so forth. In facade parsing, windows, doors and other structures should be instantiated dynamically. The displacements and relative positions are much less constrained than what is usually modeled by the DPM. Appearance-wise, the components exhibit a higher degree of variability due to different architectural styles. Even within the same building there is a normal species variation due to the wear and tear, open/closed structures, and shadows and reflections resulting form illumination changes. In addition, the single target object approach common to the DPM imposes a different problem scale than our facade application. That is to say, a face or a pedestrian can be handled in a computationally efficient manner while preserving its details. Whereas, facades will have to be severely resized to apply DPM, which will lead to the loss of intactness of geometric layout.

In terms of scale, facade interpretation is more related to scene parsing. However, scene parsing is the other extreme when it comes to being geometrically constrained. In fact, the lack of a stable layout has made incorporating context cues limited to the genre of the scene. For instance, a seaside image suggests labels of sky, sea, and sand, and eliminates possibilities of desk and keyboard. Thus, beyond the issue of existence of a structure, the localization is solely based on appearance. Another factor that sets facade labeling apart, is the size of the datasets. In object detection and scene parsing, there are datasets such as ImageNet [6] with count of training instances = 10,000,100. In facades, the available size of dataset on average is 300. Such a small size will inevitably affect the choice of the training algorithm. An algorithm that will prevent overfitting and has a good generalization

ability is needed. A distinction commonly seen in generic scene parsing that is of relevance to our applications is the categorization of regions into stuff versus things [7]. Things are objects with well defined shapes while stuff are objects with no clear boundaries and no shape prior but with a relatively distinctive stable appearance. In facades, shop, wall, roof, and sky are considered stuff.

Insight in neuroscience and human visual perception has been a source of inspiration for computer vision algorithms. One example is the current trend of Deep Learning based on Convolution Neural Networks (CNN) [8]. Its superior performance can be attributed to their reliance on the creation of a hierarchy of features that capture the image semantics at different levels of detail. This leads to handling scale space difficulties. Another factor is, the construction of data tailored filters with its improved discriminative power in classification as opposed to hard-coded kernels. Another breakthrough in DL has been in modeling by Restricted Boltzmann Machine (RBM), because of its ability to estimate joint probability of a large number of random variables. Thus, it facilitates studying the interactions among image primitives over a wide spatial scope. The extent of the investigated neighbourhood is a crucial issue in the contextual layout parsing. Weighing the benefits versus the costs in the determination of the scope size remains an active research point [9].

There are a couple of aspects commonly encountered in pattern recognition research, which we would like to clarify as it lays the basis for our work in subsequent chapters. In a discrete optimization setting which involves the selection of discrete values for random values, a prior is an important concept. It is the way to incorporate past experience about a problem. A semantic prior is an observation about a certain phenomenon that is perceived by humans and incorporated in the design of the model. One example is spatial coherence encoded in pairwise potentials in Random Fields. There are priors quantified statistically which indicate the proportion of time the random variable get a certain value learned from training data. They come with various degrees of complexity, prior distributions in Gaussian Mixture Models (GMM) [10] and 3d shape priors [11].

In statistical learning, commonly occurring classes usually exhibit higher accuracy due to the normal bias of the inference system. Whereas, rare classes with low counts resulting in low probability might be washed away by the smoothing effect. Counteracting this issue by dataset balancing

through augmentation or sub-sampling is a double-edged sword. It solves the smoothing problem but it enforces an unrealistic uniform prior on the balanced labels. Another critical point in the learning process is the balance between the training data size, the number of parameters to be learned (referred to as model complexity) and training length. The are a couple of tactics normally encountered in the literature to deal with these difficulties. Tactics include dimensionality reduction [12] and forcing sub-divisions of the parameters set to have identical values like the weight sharing in CNN [8] reduce the parameter space by several folds. When the training set size is limited, the dataset is often augmented or the pipeline is complemented with an unsupervised training phase which facilitates gathering of more samples without the need for labour intensive annotation [13].

Overfitting is one of the major issues for machine learning algorithms. It is a phase in the learning process, where error rates continue to drop on the training samples while worsening on unseen data, limiting the generalization ability of the system. The tactics usually used to handle it include composing more representative training datasets of the phenomena, by increasing the count and diversity of the samples. This is sometimes carried out by artificial data augmentation through imposing affine transformations on available images [14]. Also, overfitting is prevented by building more powerful models either with respect to the discriminative power of the classifier or the choice of the feature space. The more separable the samples, the better the classification accuracy. Such a mechanism is used in Support Vector Machine (SVM) [15] variants. Other mechanisms include the addition of dropout layers as in CNN and the stochastic approach of Restricted Boltzmann Machine (RBM) [16] learning. Throughout our work, in chapters 5, 6, and 7, we show that the use of meta-feature vectors introduces a second level reasoning about classification problems, improving the properties of feature spaces in which samples are embedded, resulting in higher accuracies.

There are different aspects by which we characterize the end model. One of which, is describing a model as parametric or non-parametric. A parametric model is one that allows to make a prediction for a new data value based solely on the system variables learned in the training phase. On the other hand, carrying out inference in the non-parametric case necessitates the existence of the training data that created the model. Non-parametric models tend to represent the phenomena more thoroughly

and they grow more with added instances to the training set. However, this comes with the expense of increased dimensionality and the subsequent computational burden. Hypothetically the two are equivalent in the case when a system behaviour can be described parametrically by an infinite set of parameters.

Another categorization is concerned with the system's ability to synthesize instances of the modeled entity. Generative classifiers model the data of the class by learning the joint distribution of observations and class from the training data. As a by-product of this procedure, the model is able to hypothesize (generate) instances of the class. At inference, the matter of estimating the likelihood becomes dependent on the probability that the unknown instance is sampled from some class. On the other hand, discriminative classifiers model the decision boundaries between the classes by learning its parameters. Thus, at inference the issue is estimating the directional displacement of the new instance from this boundary. In probabilistic terms, discriminative classifiers learn the conditional probability of class given data.

1.3 Rationale of the work

In our work, we contribute to the paradigm of contextual reasoning. we incorporate human-level cues that has lead to more automated and accurate algorithms in the field of facade parsing. Producing elucidated models of buildings is related to a wide range of applications including urban planning, conservation of world architectural heritage and creation of virtual environments for computer gaming and film industry. We advocate the coining of a new term *Topology-based Segmentation*. Topology is the gestalt effect of the geometric inter-relations between the subparts, which ultimately leads to the visually perceived arrangement. Its study is of critical importance to our problem because fitting a coherent topology entails assigning the correct inter-relations between the architectural elements. This leads to the correct localization of the elements and ultimately to the correct classification of the pixels. In other words, we are relying on layout and geometric interactions for fine-grained classification. It goes beyond location, a cue usually complementing the feature vector of the image primitive

in classification or used separately and then merged as vote source in a weighted function in a later stage. The pipeline for incorporating layout cues is standardized at a conceptual level. It starts with an initial pixel labeling based on appearance features and in a later stage incorporates contextual findings most commonly in an optimization function.

There are some challenges to implementing topology-based segmentation. Assigning a correct topological model to the scene relies on the recognition and localization of the subparts and the accurate recognition and localization itself of these subparts is dependent upon the detection of the suitable topology. This chicken-egg problem has naturally complicated the choice of the processing order for relevant algorithms between bottom-up and top-down directions. To make matters worse, the topology is vastly varied. Architecture is an art that is intrinsically dependent on creativity and uniqueness of design. Even if the style is unified, the count of the subparts and their layout is never expected to replicate a pre-seen example. More technical difficulties are encountered in the acquisition process of the images, such as the scope of the camera which balances between the details and the area captured. In addition, there is a difficulty in obtaining full non-occluded fronto-parallel shots of buildings in narrow streets and in the presence of obstacles. The aforementioned difficulties often lead to the degradation of the quality of the images and restrict the size of the collected datasets.

The reviewed state-of-the-art manifests some deficits:

- Developing hybrid systems that combine top-down bottom-up processing is widely seen as the most effective methodology to vision tasks. The ultimate goal is to able to carry out segmentation and recognition simultaneously [17, 18, 19]. However, they apply higher semantics on large-sized blobs, which we regard as a sub-optimal approach to achieving concurrency between global and local processing. Because, these approaches cause the errors from the early segmentation stages of forming the blobs to accumulate to later stages.

- To the most part, the architectural guidelines are hand-engineered rather than automatically learned from the datasets. Their parameters are even more widely accepted as being manually set.

- There is a body of priors commonly utilized by humans in the segmentation of facades that is seldom aggregated in a single optimization function. This can be attributed to the limited capability of the utilized optimization techniques or the inadequacy of the formulation of the problem. In either case, valuable knowledge is wasted in the segmentation procedure.

Our algorithms have a common philosophy in which we try to handle the challenges and drawbacks of other approaches. It is characterized by the following:

- We maintain a holistic outlook for the handled problems throughout the processing pipelines. In the assignment of labels, we alternate between local characteristics of image primitives and their global interactions. The algorithms toggle between 2 states. First one is processing image primitives while the second one is more abstract. It examines the putative meaningful models that the primitives unite to form in some latent space, disregarding the specifics of the primitives themselves. Examination of the higher-level knowledge acquired through the hypothesized models results in re-evaluation of the classification at the pixel or superpixel (resulting from severe over-segmentation) level. And the cycle repeats. In order to achieve concurrency between segmentation and recognition, the cycle should have a feedback mechanism allowing the operands of each processing stage to change their preassigned labels and affect the decisions about the operands of the other stage. To boost the effective concurrency, the toggling frequency should be as high as possible with each cycle introducing minute changes to the labeling of the primitives. Thus, the final model evolves from the accumulated effect rather than drastic sudden classification decisions.

- We minimize the use of thresholds and hard-coded parameters.In settings which involve estimating the learning hyperparameters of the classifier, we provide a scheme for estimating them in an optimization framework. Hyperparameters are those which are set prior to the training such as weights and regularization factors. Algorithmically, this permits training the classifier over all involved datasets in one go, without the need to fine-tune for each one. However, we carry out individual learning, in order not to disadvantage our methods in comparison to others.

- We use a minimal set of data-dependent heuristics in the classification process. Instead, we rely on statistical reasoning, a paradigm that has proved to be of crucial importance to vision [20, 21]. We incorporate the maximal number of priors statistically learned from the datasets.
- We follow common-sense guidelines of computation which are developing systems incurring time, space and complexity costs within tolerable limits.
- We compare our systems to the state-of-the-art algorithms and based on benchmark datasets, both in terms of algorithmic aspects and accuracy of the outcome.

1.4 Contributions and structure of the thesis

Reviews of scene parsing, object detection and CNN-based semantic segmentation are given in chapter 2. In chapter 3, we present a detailed analysis of the current and past efforts in the application of facade parsing. We show how graphical models and Bayesian inference are powerful tools for solving vision problems in chapter 4. In chapter 5, we categorize and comment on the common techniques for geometric multi-model fitting in applications such as plane, homography and motion segmentation determination.

Chapter 6 presents a novel approach to the computation of primitive geometrical structures, where no prior knowledge about the visual scene is available and a high level of noise is expected. We based our work on the grouping principles of proximity and similarity, of points and preliminary models. The former was realized using Minimum Spanning Trees (MST), on which we apply a stable alignment and goodness of fit criteria. As for the latter, we used spectral clustering of preliminary models. The algorithm can be generalized to various model fitting settings in which the spatial coherence constraint applies, without fine tuning of run parameters. Experiments demonstrate the significant improvement in the localization accuracy of models in plane and homography fitting and motion segmentation examples. The work in this chapter has been published in [22].

We propose in chapter7 a layout-based facade parsing system. We integrate appearance, layout and repetition cues in a single energy function, that is optimized through the Sequential Tree

ReWeighted message passing (TRW-S) [23] algorithm to provide a classification of superpixels. The appearance energy is based on scores of a Random Forrest (RF) [24] classifier. The feature space is composed of higher-level vectors encoding distance to structure clusters. Layout priors are obtained from locations and structural adjacencies in training data. In addition, priors result from translational symmetry cues acquired from the scene itself through clustering via the efficient α-expansion graphcut algorithm. Experimentally, we are on par with state-of-the-art. However, we make no use of dataset dependent assumptions or thresholds. The weighting of the potentials in all utilized optimization functions are estimated using the Particle Swarm Optimization technique (PSO) [25]. In addition, we are able to fine tune classifications at the superpixel level, while existing methods model all architectural features with bounding rectangles. This work can be found in [26].

Another algorithm that provides a pixel-wise classification of building facades is presented in chapter 8 that exploits Deep Learning (DL) machinery. Based on appearance, the most likely label is obtained through applying deep convolution networks. This is further optimized through Restricted Boltzmann Machines (RBM) [13], applied on vertical and horizontal scanlines of facade models to impose layout. Learning the probability distributions of the models via the RBMs is used in two settings. Firstly, we use them in learning from pre-seen facade samples, in the traditional training sense. Secondly, we learn from the test image at hand, in a way the allows the transfer of visual knowledge of the scene from correctly classified areas to others. Experimentally, we are on par with the reported performance results. However, we achieve this without specifying any hand-engineered features that are architectural scene dependent. The work is presented in [27].

Chapter 9 concludes the thesis and provides an insight for future research directions.

Chapter 2

Generic Semantic Segmentation

2.1 Overview

In this chapter we review techniques used to partition images into meaningful parts. In the early days of computer vision, the subdivision was carried out to produce color coherent regions. This class of algorithms include waterhsed [28] and mean-shift [29]. Due to real-life illumination and color variations, they produced highly fragmented regions that were of little use to tasks of object detection and scene parsing. The process often involved adhoc post-processing steps. A further enhancement was achieved by focusing on boundary detection techniques such as snakes [30] and level sets [31]. However, the basic assumption that an object is enclosed by a contour obtained from edge maps is once again challenged by the imaging conditions. Normalized graphcut algorithm [32], introduced the spatial aspect in the segmentation process by incorporating smoothness as a low level layout prior. Low level in the sense that it is concerned with direct spatial neighbours while any global optimality is achieved indirectly through propagation via tangent pixels. In later work [33], the datacost which is the penalty of assigning a pixel/superpixel to a certain subdivision became a classification score. The score is determined via a trained classifier that learned the visual attributes of such regions from preseen examples. Popular local descriptors include Cuboids [34], HOG [35], HOF [36], SURF [37] and SIFT [38]. This is how meaning (or semantics) is cast over the regions to transform them into

sub-parts of known class. In the following, we identify approaches of semantic segmentation that have been used recently in generic scenes, and are perceived as the state-of-the-art. We also review classic approaches of scene parsing that combine visual attributes with layout priors. Scene parsing and semantic segmentation are interchangeable terms. However, semantic segmentation is regarded as the more recent less ambiguous term. There are a couple of related terms such as pixel labeling, object detection/localization/segmentation and instance segmentation, which differ in the primary focus and the processing direction (whether top-down or bottom-up). However, the emergent result is more or less the same; a subdivision into labeled areas.

Despite the fact that object localization are often confined to the case of single Region Of Interest (ROI) such as pedestrian or face, it is included in our review because it can be adapted to full scene interpretation by running the same algorithm sequentially over the scene while varying the object class. Evidently, it will suffer from computational inefficiency and the drawback of searching for each object in isolation. In addition, some of its approaches explicitly formalize the modeling of geometric interconnections between object parts. Conceptually, this can be extended to a scene by modeling it as an object and its contained elements as the parts. In fact, humans' perception of part versus whole object is a subjective matter governed by scale and the required level of detail.

2.2 Scene Parsing

Most algorithms in this category involve non-parametric processing [39, 40, 9], in which the scene under investigation is compared against a repository of categorized scenes. The pipeline includes a retrieval step of a set of similar images, classification of query image and an energy function combining the classification cost as a dataterm with structural priors. Another important characteristic of this paradigm, is the incorporation of meta-learning, where preliminary classification scores are used to induce the training of more classifiers in subsequent phases. Energy is often expressed as a Boltzmann distribution.

In [39], Yang et al. apply their scene parsing algorithm to large scale problems with hundreds

of possible labels and a severe imbalance in the count of different classes. They define rare classes as the ones which occupy the tail of the frequency distribution established over the training set. In a 2 step process, the algorithm is launched with the retrieval of related images to the scene at hand from a repository based on a spatially constrained Bag-of-words image similarity measure [41]. The bag-of-words include features of SIFT [38] and RGB color. The union of labels in the retrieved exemplars provide pool of possible labels. In a 4-connectedness MRF labeling framework, they assign the superpixels of the query image to the available labels. The datacost is calculated based on a normalized intersection kernel between the superpixel set of features and its nearest neighbours superpixels in the retrieved images, in addition to an SVM classification cost. The set of superpixels of rare classes in the retrieved images are supplemented with the centroids of the classes in the training set to enhance its representation. The SVM classification is based on a local descriptor holding SIFT, RGB, location and PHOG features for the examined superpixel and its dilated encompassing region. The second step of their approach relies on constructing a global descriptor to re-run images retrieval and re-evaluate the local descriptor of the superpixel datacost for a second iteration of the MRF optimization. The global and local descriptors are based on the preliminary likelihood maps through a max pooling operation for the whole image and the direct neighbourhood of the superpixel.

The image retrieval step allows the algorithm to focus on a subset of labels which has both computational efficiency and accuracy rewards. However, in [42] the authors bypass the image retrieval step to avoid the early loss of relevant labels not instantiated in the retrieved set. Instead, they go for a classification based on the labels in the datasets. They improve the classification accuracy by fusing likelihoods from 3 Boosted Decision Tree [43] models learned from different versions of the dataset. The rationale is varying the balance ratio of the classes in the dataset so that with each version either the accuracy of the rare or abundant classes is boosted. This reduced correlation between learned classifiers was found to enhance the combined overall estimated likelihood. The combination function is a weighted summation of the individual scores, where the weights are learned from the training set as the normalized sum of likelihoods of all classes. They carry out 2 iterations of MRF optimization. In the second run, they add a weighted global context cost for the labels. The cost reflects the frequency

of the selected set of labels obtained in the first round, after expanding the set with other labels. The added labels are the ones that share images with the obtained set. The frequencies are calculated over the training set.

In their work [40], Tighe et al. handle the case of overlapping instances of the same or different classes by producing individual clear boundaries of the objects in contrast to the norm of producing a single blob for tangent instances. Their dataterm is the result of a sigmoid function applied on an SVM score. The input to the SVM is compounded from a vote score from pixels in the retrieved set similar to the pixels under investigation in addition to a score from an object detector. Thus, for every pixel in the image, they run 2 specialized SVMs. One of them is for the detection of occluding objects (things) and another for the occluded ones (stuff). Validity of the occlusion relations between objects is verified based on a criterion on histograms of overlap between groundtruth object polygons in the training dataset. Their final inference step is solved via an integer quadratic programming optimization function that takes into consideration overlap constraints.

The method of [9] sub-samples rare and abundant classes with varying intensity such that more samples are discarded from the abundant class to achieve balance. The algorithm works at the superpixel level after subjecting the image to an over-segmentation by the SLIC [44] algorithm. Superpixels are represented by a meta-feature vector. The raw features of dense SIFT [38], LBP [45] and RFS filter banks[46] are used to build clusters . Each superpixel is then represented by the histogram of cluster indices assigned to its neighbours, based on centroidal distance. The vectors are categorized according to the image zone to which the superpixel belongs. They are then used to train local SVM classifiers, one for each zone. A crucial point here to note is the choice of the zone size. They needed an approach to overcome the drawback of using absolute location, which is subject to misalignment resulting from varying viewpoint and/or scale. Firstly, at test time multiple SVMs of zones near the superpixel are called for classification and their voting is averaged and smoothed further in a Condition Random Field (CRF) model. In this way, they modulate the impact of zone subdivision. Secondly, they perform an elaborate procedure to select the zone size. They carry out a bias-variance tradeoff [47] based on varying the neighbourhood scale and measuring the KL-divergence between

the target and the learned model. The optimum zone subdivision is the one that minimizes the testing loss which is the addition of the bias and variance. Also, [48] provides their final classification in an Markov Random Field (MRF) framework. The dataterm is a negative-log function of an initial likelihood compounded with classification posteriors from specialized detectors of specific scene structures, namely porous/solid scene elements and vertical/horizontal lines. The piecewise smoothness prior depends on adjacencies on sky/ground separating line, vanishing lines and intersection lines between planar surfaces. The initial classification is obtained via the algorithm proposed by Hoiem [49]. It merges the classification likelihoods from 3-class random forest classifiers trained and tested on different segmentations of images resulting from SLIC [44], FH [50] and CCP [51]. Then, they use the Hoeim [49] approach to expand labeling to the full set of 7 classes corresponding to finer-grained scene structures.

The work of [52] explains that scene layout when embedded in the 3d space, provides more reliable neighbouhood relationships that are invariant to viewpoint variations. Their framework works in both 2d and 3d space. The included algorithm operates on a dense depth map obtained via guided depth enhancement technique [53] performed on LiDar image pairs. From the point cloud, they take out the estimated ground plane points obtained by RANSAC [54]. Using k-d tree clustering algorithm, they produce groupings of points corresponding to object proposals. Features of 3d location and RGB color are considered for the seed proposals. And a Gaussian Mixture Model (GMM) is built for each object, with special handling for the sky class as it can not be sampled from LiDar images. A Convolutional Recurrent Neural Network (CRNN) [55] is fed with each 2d image region corresponding to the 3d object hypothesis yielding a semantic label for the patch. Then, they use an elaborate CRF energy function with the aim of producing a semantic final label in addition to an object instance index. Unary potentials come from the likelihood of an object based on Gaussian mixture parameters in 3d space and the CRNN score combined with geometric constraint that enforces the objects to lie on the ground plane except for sky and background. Piecewise smoothness is encouraged on the object and category level depending on features dissimilarity. Also, a coherence penalty is added if the 2d category label does not match the recognition result of the 3d object detector.

An example of a hybrid system that produces the final classification of pixels based on the summation of energies coming from a local and global view can be found in [56]. Firstly, the image is subjected to an ensemble of 4 CNNs, whose predictions are obtained through maximal margin inference performed on the CNN features. Each component CNN in the ensemble is fed with image patches of unified label. The discriminating factor between the 4 CNNs is the sampling method used to overcome the severe imbalance problem in the scenes. They correspond to global sampling, class sampling, hybrid sampling and truncated class sampling. In this way, they were found to complement the properties of each other resulting in improved accuracy figures that can be attributed to an increased differentiating power among rare classes. The yielded classifications are averaged. The global vote for the pixel classification comes from a weighted voting of the k-NN pixels in exemplar scenes retrieved from the training set. To this end, a holistic feature vector is constructed through a pooling operation on the resulting features of the CNN applied on image patches. Exemplars are retrieved based on a dissimilarity defined on this vector.

Pruning the pool of labels help in resolving labeling ambiguity for mutually exclusive classes. Exclusive in the sense that they do not coexist in images of the training set. One can find that a global scene genre resolves ambiguity between grass and a desk in a landscape scene. Obviously, this kind of reasoning can not recover the true label if the dilemma is between 2 coexisting classes such as grass versus sky. Generally speaking, context in current scene interpretation algorithms is utilized to indicate that the existence of some objects in the image either promotes or suppresses the detection possibility of others based on co-occurrence priors. However, the utilization of context in the problem of object localization is fairly rare.

2.3 Object Localization

The sequential search is the technique found in several facade parsing systems, in which the algorithm iteratively scans the image for all possible types of objects. Examples include [57, 58]. Because architectural elements are not structurally versatile- all of them can be fairly approximated with bounding

rectangles, object detection in this respect is confined to appearance features. Object localization is mostly cast as a problem of finding the encompassing Bounding Box (BB). This leads to sub-optimal labeling for highly interleaving and concentric objects. One of the earliest techniques in detection of objects was matching between object templates and the images in a sliding window manner. Matching to a template was based on photometric properties of the objects. There is a growing interest for incorporating contextual properties in the search. Positions of the sliding window with *high* correlation scores indicate the target presence. Thresholding was the obvious mechanism for quantifying the subjective description *high*. Due to the limited ability of templates in capturing real life variability of objects, pictorial structures were introduced [59]. They had higher flexibility in encoding object appearances. In addition, they introduced a very fundamental idea, which is making use of the inter-relationships among object sub-parts to boost the recognition accuracy of each object. The inter-relationships were modeled as spring-like connections. The metaphoric "stretching and shrinking" of the springs increases the model flexibility to accommodate different geometric layouts. It laid down the conceptual basis for ideas like and Deformable Parts Model (DPM) [5, 60] and its variant constellation models [61].

The authors of [62] propose localizing objects by studying the spatial and appearance similarity relationships among a set of putative bounding windows resulting from [63]. The algorithm forms a vector of meta-features that are used in calculating a goodness score for each candidate window. Their main contribution lies in the choice of of the features. The meta-feature is designed to boost the selection of windows that exhibit spatial relationships to other candidate windows conforming to a distribution predicted through a GPR [64]. The distribution is learned from the collection of candidate windows for samples in the training dataset and their relationships to the groundtruth window. Understandably, it encourages windows with the least uncertainty about the prediction based on characteristics of the training set object category. For each window, the features include an all-pair appearance similarity measure with other windows weighted by the discrepancy between their spatial displacement to the reference window. Spatial relationships are determined based on the 3 topological aspects of overlap, part-of and containment. The weighting of the features is obtained through

structured output regression formulation [65] solved using quadratic programming with constraint generation [66]. Negative weights are allowed and learning is optimized for each object class. The scoring function is efficiently handled by eliminating the calculation of all-pair meta-feature for all windows by the early elimination of windows that do not improve the upper bound on the score.

Motivated by the urge to boost the recall figure of the algorithm, even at the expense of collecting more false positives and lowering precision, [67] relaxes some constraints in the initial object detection phase. This is done to overcome the problem of early rejection of correct object suggestions. They apply a standard object detector that produces BB and a viewpoint proposal. They lower the rejection threshold by passing a non-maxima suppression kernel and expand the proposals by using selective search [68] and edge boxes [69] methods to include non-class specific boxes obtained only through an objectness measure [70]. In addition, they generate a set of 3d object boxes resting on an extracted ground plane. The final object detections are sampled from a distribution of pairwise and higher order density functions established from the training set. As such, the original proposals are considered as seed points that direct the subsequent search for final detections relying on the learned contextual relations. Pairwise relations encode relative location and orientation. Whereas, the higher order ones are formed through defining a bank of words based on the pairwise relations after being discretized. Then, they apply a customized topic modeling [71] using latent Dirichlet Allocation [72] to discover the most common arrangements for objects with their likelihoods.

The search for an object in discriminatively trained parts-based models [5] entails convolving the model parameters with image ones at various positions and choosing the position which results in the maximum score. The parameters are obtained after transforming the image RGB color to another more reliable feature space such as HOG, SIFT...etc. The image is multi-scaled to obtain the features at different resolutions. The model parameters include a root filter corresponding to a BB of the whole object. This filter is augmented with sub-part filters. Deformations in the sub-parts are deducted from the convolution scores. They are expressed as deviations in the optimal positioning of sub-parts relative to the root. To accommodate for more inter-class variability, each object is represented by a mixture of component models. In the training, the collected positive samples of

each class are roughly subdivided into components comprising the mixture model. During training only the location of the object BB is manually specified, while the optimal placements of parts are considered hidden variables learned via special SVM formulation called latent SVM. It selects the highest scoring location of parts of each positive example. By positive, we mean a true detection. The count of part filters is predetermined and their locations are initialized in the training to blobs of high response when convoluting root filters. The star configuration utilized in these models lacks the ability to learn the positioning of sub-parts relative to each other, only to a root node. At testing time, one of the common approaches to decide the positions at which the model is tested by using an interest point detector [73, 74]. Obviously, the cost function of the DPM has a dataterm encoding appearance penalties and a spatial prior for the geometric relationships. Inference in DPM is normally solved by dynamic programming. When determining the anchor placements of the parts, the correlation scores of part filters are spread within a neighbourhood of recorded position to allow for tiny displacements, while taking into consideration the prespecified displacement costs. To enhance the training process and form a more balanced dataset, the algorithm only retains hard negative samples in a cached subset via bootstrapping. Negative examples refer to the instances which are incorrectly classified so far. The subset is updated iteratively with the evolution of the classifier parameters. When dealing with mixture models, the concern is how many components are required in the bundle to represent a certain category. Intuitively, the number is directly proportional to the broadness of the variability spectrum within the category and inversely proportional to the expressive power of the single model. That is to say, the better the generalization ability of the individual model the less need to include more components in the mixture. Also, we would like to point out, grouping the models into mixtures is a conceptual step. Computationally, all models in a mixture are tested on the image in the same manner as models from different mixtures.

In [75], the authors extend the deformable parts-based configuration to include random variables representing both local and global context. The scoring function now takes into account 4 contextual classes occurring top, bottom, left and right to the root BB; with their deformation again represented as the displacement from the anchor positions. The feature vector of the contextual pairs is comprised

of a normalized pixel count of all considered classes within a window. Also, they consider in the configuration score, a binary global feature vector indicating the presence of each class anywhere in the image if its pixel count exceeds a certain threshold. They choose to obtain the preliminary semantic segmentation results using the O2P algorithm [76]. Basically, it classifies superpixels of the image based on their SIFT features. Their enriched DPM is tested on Pascal VOC [77]. They provide a pixel-wise labeling of the dataset and raised the number of included object class categories from 20 to 59. The matching of the trained DPMs to objects in the scene leads to a complete semantic segmentation of the images.

[61] designs an algorithm to learn the model's parameters in an Expectation Minimization (EM) framework with the involvement of the A$*$ space search algorithm to accelerate the the evaluation of the likelihood of a hypothesized model. [61] is a generative approach characterized by the elimination of non-object samples in the training. The occlusion issue is explicitly handled in calculating the scoring function for an object hypothesis by incorporating a binary vector encoding absence/presence of parts. Objects are tested at certain locations in the image called features. These are suggested by Kadir and Brady feature detector [78]. It selects a subset of local maxima of an entropy function defined on a histogram of circular regions. They estimate the scale from the relative size of image features. The scale is then used to parametrize other aspects of the algorithm. They specify visual descriptors of regions by a PCA dimensionality reduction on raw image patches.

A top-down bottom-up hybrid is provided in [79]. Cadena et al. use a dual approach that relies on individual object detectors (similar to object bank [80]) and a semantic segmentation module overlaying context to the objects. They use VM-based algorithm [81] to obtain pixel-wise probability maps. The maps are used to build the feature vectors for the BBs yielded by an ACF object detector [82]. For each candidate detection, its BB is dilated and translated in various directions and a normalized average of probability scores of the resulting BBs are concatenated in a single semantic vector. The idea is to capture the local context of the candidate detection. The semantic space vectors are scored via a SVM with a kernel based on Bhattacharya distance. The resulting score is back mapped to the pixels and combined with the ACF scores and a vote from a shape prior evaluated per pixel to

obtain a final classification probability. The prior is constructed through rescaling and accumulating the probability of the groundtruth object boxes and performing a logistic function on them. It is an examples of a meta-learning that combines voting from various classifiers.

The work in [83] is another example of a system that merges inspirations from object parts models and scene parsing, that is relevant to our application. It explicitly models the interactions between scene structures. It overcomes the limitation of the star topology by using a Scene-Object Graph (SOG) to represent the structure of the scene. SOG is able to to encode inter-relationships between pairs of parts, not only to a root node. It is used to store the latent topology of a specific scene category. It is restricted to a tree structure to avoid loops and allow efficient optimization of the topology through the expansion of a weighted minimum spanning tree. Nodes of the SOG hold appearance models of the objects that are prominent in the scene. A criterion that is dependent on the frequency of occurrence of an object across the category instances. In the same manner, edges are added only if they signify persistent relationships among objects. The ultimate aim becomes the discovery of the most relevant SOG to the scene at hand thus leading to identification of scene category. To this end, the algorithm searches for the SOG that best aligns with the image with minimum deformations in relative positions, expressed as a distance transform function. This inevitably yields a subdivision of the image into object bounding boxes. The approach suffers from 2 shortcomings. Firstly, BBs are not tight. Therefore, there is overlapping between different object boxes and the same pixel can be multi-labeled. Secondly, the SOG is a static graph with no ability to get updated dynamically to accommodate new instances of an object category.

2.4 CNNs

CNNs are known to excel on data that manifest frequency variations coupled with a locality property in some space. This makes them perfectly fit to take in images of real values, without the need for preprocessing. CNN dependent algorithms occupy the leading positions in any performance ranking on benchmark datasets for semantic segmentation such as Pascal VOC 2012 [77]. The basic idea is

the network is allowed to learn the parameters of specialized filters adaptively. In [84], the authors show that CNN is basically an extension of the Deformable Part Models (DPM). CNNs belong to the paradigm of deep learning in which a lot of layers are piled over each other to build the architecture. A well-known problem that has long hindered learning through backpropagation was the vanishing/exploding gradients, where the error signal drops/shoots drastically leading to unstable learning. One approach to solving it was the introduction of standard Gaussian normalization layers for the inputs of activation layers [85], besides normalizing the initial weights and biases of the network. Another aspect that allowed architectures to get deeper than traditional Neural Networks (NNs) is the concept of weight sharing. It substantially reduced the number of parameters per layer to trivial sizes allowing the efficient iterative update of the parameters through the backpropagation algorithm. Weight sharing achieves translation invariance by suppressing the discriminative power of location. Also, it increases the expressive power of the filter as it is trained on a number of patches much larger than the number of images originally in the dataset. Using a small-sized translation invariant filter has been possible due to 2 observations. Firstly, the classification of a pixel depends mainly on its neighbourhood based on the local smoothness prior. Secondly, regions belonging to the same semantic tag share a near stable configuration of appearance statistical properties regardless of their location in the image. One can not overlook the role of GPU hardware in the implementation of such massive networks. It allowed the concurrency in the processing of large sized images making the training computationally feasible.

A typical architecture is a repetition of an entity composed of 3 layers; convolution layer, non-linear and pooling. On top of which, the architecture is sealed with a logistic loss layer that calculates the deviation of the resulting prediction with respect to the required from the groundtruth. The deviations are propagated backwards through the chain rule.

Convolution layers perform successive linear transformations of the input layers. Non-Linear transformation is achieved through activation functions. Traditionally, they were the sigmoid and tanh functions. Non-linear Rectified Linear Units (ReLU) (equation 2.1) proposed by Nair and Hinton [86] was found to converge to the same levels of training errors at a fraction of the iterations. Theoretically,

ReLUs donot need normalization of their immediate inputs. Pooling is basically spatial sub-sampling. It increases the receptive field of subsequent filters gaining a wider overview, lowers computational period and achieves smoothness in segmentation. In case of image recognition, fully connected layers are added to provide the holistic features. In the loss function layer such as cross-entropy, the objective is maximizing the log-probability of the correct label under the prediction distribution.

$$f(x) = \max(0, x) \tag{2.1}$$

A breakthrough contribution was presented by Krizhevsky et al. [14]. They proposed a cascade of layers that achieved unprecedented results on the ImageNet dataset and in the ILSVRC-2012 competition [87]. For the input, they add principle components to the RGB channels to provide robustness against real-life variations in color intensity and illumination.The authors state that their enhanced performance is mainly due to the extensive tactics they used to overcome overfitting. The included layers were the aforementioned ones in addition to dropout regularizer layers [88] that were proved to reduce overfitting. The idea is to suppress a set of neuron responses to 0 stochastically. Thus, a different response architecture is propagated through the layers. The effectiveness of their cascade can also be attributed to a brightness normalization step applied to adjacent kernel maps in a certain locality produced by the ReLU units. In addition, their overlapped pooling scheme had favourable effect on overfittting.

Deeper CNNs followed, such as VGG-16 and VGG-32 [89] and ResNet [90]. In ResNet, they densely add shortcut connections in a periodic manner that bypass a block composed of linear/non-linear/linear layers. These connections allow the network to learn an alternative function composed of an identity mapping added to the residual function. The alteration resulted in more efficiency in the learning process, allowing the mounting of more layers. A favourable effect detected in residual networks, is the reduction in magnitude of layers responses resulting from the diminished share of each layer, when more layers can now be inserted. Also, the phenomenon can be rooted to forcing the weights in the direction of zero values. These networks are generally acclaimed for their fast

convergence. Maxout is another type of layers that are added to solve overfitting problems with the explosion of number of learned parameters. In [91] based on the studies provided included, Zagoruyko and Komodakis claim that the effective depth of the ResNet is much less than the reported due to the inability of the gradients to be propagated beyond a certain number of residual blocks. Thus, they conclude that the ResNet is basically an ensemble of shallower CNNs and that the network can perform equally well with a much shallower architecture.

Most of the reviewed architectures focused on the properties of the mounted layers that characterized the learning in CNNs. They include the dropout rate, down-sampling via maximum/average pooling or by increasing strides, whether batch normalization is implemented or not, the schedule by which learning rate is changed, and momentum and weight decay parameters. On another front, significant efforts have been made to boost the use of the GPUs such as the parallelized implementation on multiple GPUs in AlexNet [14] which necessitated a different connection pattern among the layers. The disconnected columnar configuration in [92] is another example of such outlook.

Lately, these networks have been adapted to the task of semantic segmentation. Earlier efforts include [93, 94, 19]. They have focused on CNN as feature extractors and the final per-pixel classification was left to other techniques. In the more recent studies there is a direction to replacing the $1 \times m$ softmax layer with an $r \times c \times m$ classification layer, where r and c are the dimensions of the image and m is the count of available labels. In other words, the fully connected layers are replaced by $1x1$ convolutional filters thus allowing end-to-end training. In addition, deconvolution layers reverse the pooling effect, coupled with an interpolation layer to restore image back to its original size. The breakthrough work in this respect has been presented by Long et al. in [95]. Other characteristics include skip connections that allow merging of features from widely apart non-subsequent layers of the CNN corresponding to local appearance features and higher level semantics, to allow predictions to be made at fine grain level without losing the more global outlook. They use upsampling through the so-called deconvolutional layers to map predictions to original image size. Deconvolutional layers perform reverse convolution with a bigger output stride than the input stride. In other words, a single input is connected to multiple outputs. In some layers, the deconvolution kernel is fixed to a bilinear

interpolator. In others, the filter is again learned adaptively based on the backpropagated gradients.

The authors in [96] try to solve the coarseness problem of the FCN output, due to the sudden upward resizing of the feature maps which happens in top-most layers. They add deep deconvolution layers that exactly mirror the original convolution network, giving a butterfly shape to the overall network. This leads to a more gradual upsampling of the feature maps and subsequently the predictions. The deconvolution concept is handled in a different way. Instead of learning linear filters of [95], the network stores pixels locations in the downsampling. In upsampling, it propagates the obtained features of each pixel to its neighbours. A similar approach to resolving competing activations was used in [97]. The network is run for each hypothesized object in isolation. Later, they are merged by maximization of values to form final scoring maps. They also differentiate between 2 levels of training. In easy level, the network is applied on bounding boxes that totally encompass objects. While, in the more challenging level, it is supplied with BBs that overlap object instances with varying degrees. This accounts for the case where the subdivision at inference time breaks down whole objects. The authors explain that their architecture is highly dependent on batch normalization to maintain efficiency.

In [98], the authors define hypercolumns as a stacked concatenation of convolutional features obtained per-pixel throughout the different layers of the network as a way to capture different levels of abstractions in a similar manner to skip architectures in FCN[95]. They use location specific linear classifiers in a $k \times k$ grid-like arrangement at training time, while at testing they interpolate the results of the k^2 classifiers when applied on the whole of the image. A similar outlook can be found in [99] where CNNs are trained on image patches and at inference results from different patches are merged using the sliding window.

Kampffeyer et al. [100] apply an ensemble of CNNs on remote sensing images. The ensemble consists of a patch-based CNN and a pixel-based one. They carry out a data augmentation procedure to increase the size of their small dataset, which is originally 33 images. Their networks are on the shallow side approximately 6 convolutional layers. To counteract the imbalance problem, they utilize a median frequency based weighting for cross-entropy losses. The ensemble scores are merged into

a final classification using one-vs-all linear SVMs. In their experiments, they investigate the effect of thresholding Monte Carlo uncertainty maps [88] of the ensemble on the accuracy figures. In [101], the authors utilize a Convolutional Feature Mask (CFM). Object proposals are obtained through selective search [68]. It maps the proposals to regions with the highest response in feature maps after receptive field adjustment. The resulting proposals are then fed individually to fully connected layers for recognition as a whole. As for areas occupied by "stuff", they carry out a similar procedure but customized by a segment matching pursuit similar to [102] to select the most compact mask for the stuff region.

Newelle et al. [103] employ a 2-phase hour-glass pipeline in which the first phase operates as usual, whereas the second is fed with the preemptive predictions. In essence, the CNN is learning a mapping between the erroneous scores and the true classifications. However, the fact that the error is backpropagated from the objective layer of the second phase to the input image of the first phase makes the value of the intermediate mapping questionable.

Context as a cue is present when mounting convolutional layers with a filter size of j but is confined to a local neighbourhood. The receptive field size at layer t becomes $1 + \frac{j-1}{2} \cdot t$. This is further boosted by the pooling layers. More explicitly, [89] makes use of fully connected layers which provide a global investigation of the latent representation of the image, at higher levels. Chen et al. [104] use a special sparse *a trous* filter which spans a wider area of the scene. In [105], the authors allow varying the parameters among the convolution filters to build ones that are specific to each image zone. This contradicts the design principle of weight sharing, but is expected to improve accuracy, when there is an evident affinity of scene components to certain image coordinates.

Chapter 3

Facade Semantic Segmentation

Generating models of buildings has gained considerable attention in research [106, 107, 108, 109], due to its importance in innumerable applications, such as heritage conservation, disaster management and urban planning. One particular field of interest has been analysis of building facades. In this chapter, we review past efforts in building facade interpretation [57, 110, 111, 112].

3.1 Feature vector classification

In this category the focus is employing the most powerful classifiers without much attention to the semantics of the application at hand. In [113], the learning proceeds by recursively partitioning the training data using decision trees. Intermediate nodes are called decision stumps and they direct the formation of the data sub-clusters based on a mutual information criterion, while ultimately reaching data of a uniform label at the leaf nodes. They advocate the use of a Gaussian process classifier at the leaf nodes which is merited by its ability to model the latent function that maps an input feature vector to an output label without static parameters. It also solves the optimization analytically in a tractable manner in the binary case without the need for approximate inference. The Gaussian process is adapted to the task of multi-class labeling through a one-vs-all scheme. They use a Monte-Carlo technique to sample the gaussian smoothed posterior maps of preliminary segmentation obtained

through mean-shift on Opponent-SIFT [114] over 5 scales of the image.

[115] also makes use of impure decision trees, where each leaf node holds probabilities of multiple classes. Learning in such trees is about deducing the splitting decisions which involves choosing the node to be split, the attribute which controls the decision and its thresholding value. Choosing the node is usually based on an impurity measures which indicates the frequency of occurrence of different classes at some node as in the Gini-coefficient and mutual information criterion. The constructed trees are improved through the Breiman procedure [116] which converts intermediate nodes to leaf nodes if they add complexity without reducing misclassification significantly to prevent overfitting. They utilized features selected in in [117, 38] and their so-called *contextual priors* are actually a prior expectation of the classes found at intermediate nodes.

In an empirical study [118], the authors compare an extensive list of feature sets used to facade pixelwise classification. The sets constitute of RGB and HSV color properties, statistical measures of histograms defined on gradients, moments and eigen values, textural features derived from Walsh transform, SIFT features, and geometric properties of regions (area, perimeter, compactness and aspect ratio). The regions were yielded from the mean-shift algorithm. They report highest accuracy from the histogram of gradients.

The algorithm [119] starts by proposing a set of regions as an outcome of an unsupervised segmentation method. Then, it creates a feature vector of structural information but on a region-basis in contrast to the pixel-basis vector of [120]. The features include more or less the same set of features of [118] combined. Such a mega feature vector would normally require feature selection to minimize outliers, but it was skipped.In a separate postprocessing step, the classification is refined in a CRF framework solved with α-expansion [121]. The unary encodes the classification posterior complemented with normalized location information. The pairwise potential encourages non-coherent regions to be assigned to different architectural elements. This is uncomprehendable to us since the submodularity constraints and the zeros on the diagonal of the edge matrix [122, 123, 124] of the α-expansion would only encourage same labeling on whatever basis.

Other pixel-wise merit functions are only applied on appearance qualities, lacking the layout

perspective in the classification such as[125]. [120] is the only reported work that allows a per-pixel final classification while incorporating layout cues in this category. Every pixel is represented by a vector of image features (such as: location, RGB values, and HOG features), in addition to contextual ones (such as: neighbourhood statistics, and bounding box features) obtained from the preliminary predictions based on image features. The drawback is, each feature vector is supplied independently to an ensemble of classifiers. It lacks the concurrency in classification of pixels of the arrangement as it does not model the interaction between image primitives. Hence, it lacks the global optimality in the proper sense.

3.2 Expert-based layout enhancement

The work complements the feature vector classification by imposing a set of architectural guidelines that are manually designed. They are more flexible than scene grammar [126] and aims to ensure integrity of the layout beyond what is perceived from appearance cues. These guidelines are concerned with alignment, symmetry, similarity, co-occurrence and components layout. In [57], Martinović et al. make use of these architectural principles in their final classification decision. They refine the output of a preceding segmentation step by applying this set of restricting principles in an adhoc procedure. Each principle is applied in isolation and in most part, as a matter of fulfilling a certain criterion is exceeding a manually specified threshold. The segmentation is presented as a classification problem, in which each pixel is assigned a likelihood for belonging to a certain semantic structure. This is achieved by an RNN [127] fed with an oversegmentation of the image and a Dollar's Integral Channel [128] specialized window and door detector.

3.3 Parsing grammars

Formal grammars are one popular approach to facade parsing [129, 130]. Inverse procedural modeling require a set of parsing rules to carry out semantic segmentation. The rules constraint the arrangement of architectural elements. The grammar is context-free and consists of a set of production

rules that recursively converts non-terminal variables into others until reaching terminal variables corresponding to architectural elements. The derivation takes the form of predefined operations of horizontal/vertical splits yielding elements placed in bounding boxes. The production rules are often accompanied by parameters that indicate the specifics of a certain derivation.

In Teboul's work [110], the split grammar produces only binary trees. The split is either vertical or horizontal. Terminals specify bounding rectangles with their structure label, position and dimension. To make the optimization tractable, the authors enforce constraints on the production rules such that only non-terminals on the right hand side are cyclic and they include mutually cyclic rules. They formulate a Markov decision process where the transitions are deterministic because the change of state corresponds to applying a grammar rule. Through Bellman's equation, which expresses the process recursively, they maximize the accumulated gain when implementing a certain sequence of actions corresponding to derivation choices. The optimization takes place through one of the approaches to reinforcement learning, namely Q-learning. The bigger challenge for the optimization is the determination of the split parameters which specify the exact location of the split affecting size and position of the resulting structures. The putative split parameters are randomly chosen and they rely on the Q-learning ability to cut down on further computation in unpromising moves. The gain is defined by a merit function that is based on posteriors obtained through a RF/GMM classifier trained offline on RGB values of image patches.

In most cases the Split Grammars are manually designed by human experts [131]. However, more recent contributions are targeted towards automatic learning of shape grammars. The algorithms are primarily about optimizing an energy function that specifies a series of split operations. It generally operates in a fully supervised setting. Initially, the algorithm designs a set of production rules obtained from the parse trees to capture the hierarchical structure of annotated samples in the training set. The edges correspond to separating lines between elements of different structure genres. The set of rules is refined to enhance its generalization ability and its compactness by removing redundancy.

In [132], the algorithm is user-assisted in the initial phase. The processing proceeds from a simplified set of generic rules to produce the parse trees from the groundtruth while maximizing the

number of correctly labeled pixels. The rules obtained from the parse trees are then subjected to rule compression and clustering. Compression allows to remove redundancy resulting from repeated subtrees with the same parameters and structures. The search for repeated subtrees is efficiently carried out using Valiente' algorithm for isomorphism [133] that relies on accumulating the subtrees in hash tables. Specifying the parameters associated with the meta rule boils down to assigning the average of all positions encountered for this rule. They go on carrying a LP-based clustering algorithm [134] that groups together similar subtrees across different image instances. The LP algorithm aggregates based on the distance of each tree to the centroid of the cluster while favouring trees of more depth as centroids. The distance is based on structural and parameters similarity of parse trees. Merging is actually carried out by the similar trees with a couple of derivation rules equivalent to their largest common part. They choose the weighting parameters in the clustering objective by optimizing the well-known indices of DB, Dunn and global Silhouette [135, 136]. Martinovic and Van Gool [126] utilize a bottom-up approach by subdividing the image into tiles. These tile are combined agglomeratively to form the parse trees and subsequently indicates the rules. These are then refined using a MDL measure again to remove redundancy and boost generalization. In [137] Weissenberg et al. defines an energy function over split line proposals that encourages localizing the split lines over image edges of longer length and penalizes the break down of regions of greater affinity measured by the co-ocurence of assets (special edge points). We see for the first time a procedure to yield n-ary rules instead of the binary ones. However, the algorithm is utilized for procedural modeling in which facade synthesis and retrieval is the goal. They do not report results on semantic segmentation.

Kozinski and Marlet [138] represent the image as a factor graph. The graph is generated dynamically based on a parsing grammar. Factor nodes specify the objects (wall, sky, window..etc.). In a closed-loop control system, they generate a preliminary model represented as a factor gragh where variable nodes correspond to split and snap lines using the grammar which is iteratively refined based on the feedback of an energy function. There is a cost incurred for each production rule that involves its likelihood of being applied at the current non-terminal node. What characterizes the node is the sequence of its ancestor nodes and the value of likelihood of alikes are obtained from groundtruth

data. The factor graphs are converted to MRFs. The energy defined on it corresponds to the amount of deviation of the hypothesized model to the image in terms of position of the parts that assigns potentials based on violating relative positions of geometric primitives. It also penalizes depending on textural violations of the perceived bounding boxes on pixel basis. Weight parameters are estimated using max-margin MRF training. The structure label of the segments is solved independently of their sizes and positions, but both are solved with TRW-S [23]. Other work include [139].

Despite the high performance results, the generalization ability of the grammars are still questionable and the fact that part of the optimization function is defined over a continuous domain of size and position parameters downgrades the efficiency of the parsers. Also, they often require perfect alignment between structures and the designated set of rules is highly style-specific.

3.4 Repetitive patterns

In the grammar-free paradigm, the cue of repetitive patterns prevails. Symmetry as a generalization of repeated structure, is considered a pre-attentive feature that enhances the visual perception of humans. Architectural design exhibits translational symmetry, which is a type of transformation that preserves patterns and subsequently enables the establishment of correspondences. Most presented algorithms are usually applied on perfect lattice-like arrangements, which manifest high degrees of symmetry with substructures repeated with high frequency (figure 3.1). There are other cases where the repeated structures exhibit no evidently dominant lattice structure. In these cases the grouping is only caused by similarity in appearance and should be flexible enough to accommodate various geometric layouts and low count of points in support of the grid. Some authors assume a fixed dominant direction. A restriction that could be relaxed in algorithmic sense by leaving translations unconstrained but will incur a substantial computation burden.

In [140], the authors perform a grouping of feature points descriptors. The descriptors are combined vectors of lower level image features, namely, KLT, SURF and MSER. This is to cover a wider range of visual elements. Since the number of possible repeating intervals is not given a priori, they

(a) (b)

Figure 3.1: A sample of 2 images illustrating the difference between (a) a proper lattice and (b) repeated structures.

use the mean shift algorithm to perform the clustering. For each cluster of descriptors, they generate a spatial configuration from a *quad* of points on an integer lattice basis and transform the rest of the image points by the same perspective transform. If the quad represents a valid salient 2d lattice, then it will have multiple similar configurations after the global points transformations. They perform a lattice completion process that tracks the undetected lattice points depending on a normalized cross correlation between the basis quadrilateral and a rectified version of the input image. Lastly, they perform a sequential lattice grouping that depends on the degree of overlap between the proposed lattices, after sorting them according to a measure of goodness named the A-score.

In [141], Wenzel et al. extend the work presented by Loy and Eklundh [142] in the domain of symmetry detection based on matching between SIFT features. The matches are encoded by their symmetry axis in a Hessian normal form and clustered based on $2d$ thresholded histogram peaks to detect prominent regularities. The quality of the detected symmetry is evaluated based on an orientation measure. After the symmetry extraction, they resort to their containing convex hulls to determine the repeating structures in rectified images. They heuristically get the number of base translations from the histogram of coefficients of linear combinations of basis vectors to produce the

most compact representation of uni-displacement grids.

Zhang et al. [111] extract axis-aligned boxes in an interactive procedure in which a user specifies a seed structure which is then used as a basis for searching for alikes. They perform a series of lattice enhancements such as aggregation of the initial elements in grids, completion of those grids based on principles of alignment and equal spacing, and separating them in layers when fulfilling an overlap criterion. Doing structure completion in overlapping layers that occlude each other is not straightforward and requires a choice that optimizes a global function. However, they solve the problem heuristically relying on a size of grid criterion. The layers are then subjected to binary hierarchical decomposition of bounding boxes. Each layer is split into instances of architectural elements. The quality of the decomposition is quantified by their proposed integral symmetry criterion of the subdivisions and optimized by a genetic algorithm. The criterion considers intra/ inter measures of spacing and area overlaps among aligned boxes and their reflections on the vertical axis. Optimizations done heuristically or randomly through a genetic algorithm undermine the approach. In addition, even the search requires that marks placed by a user must be contained in structures similar to the seed. This makes the subsequent phases of questionable value.

In [143], the authors make use of the existence of false positive matches in an image pair, together with the matched feature descriptors between the image and the mirrored image, to establish the existence of a reflective symmetry. These points suggest the resemblance between different parts of the architectural model. These points are used to establish correspondences on the initial 3d model. These correspondences suggest transformations that are then verified using RANSAC algorithm.

In [144], the existence of repetitive structure was used to aid the estimation of epipolar geometry. First, Canny edge detector was used to detect edges which are grouped to form line segments. The image is then rectified and the differences between all pairs of intersections of the detected line segments and the axes, are recorded as possible horizontal and vertical repetition intervals. For every interval length l , they build a histogram for the remainder of the division of the rest of the intervals by l, such that l and its multiples are considered as one choice. The 3 highest scoring intervals are selected for each image. The score is dependent upon a histogram derived measure and directly proportional to l.

The work in [145] provides an example of the use of lattice structures to enhance the process of urban reconstruction from sparse 3d point clouds acquired through a LiDar scanner. The process is initiated by the manual determination of the SmartBox, which loosely defines the extent of the repeating block. The user presents a sequence of SmartBoxes through drag and drop operations, from which the expected length of the repetition interval can be deduced. An automated optimization operation called SmartBox Snapping is used to refine the position and size of the boxes. It balances between the data fitting cost to the input points and contextual constraints that regularizes the inter and intra box relations, under the Manhattan world assumption. The data term favours the fitting of the facets and edges of the box to relatively close and densely sampled uniform data points. As for the contextual term, it measures how well the interval length between the initial and the examined SmartBox matches the expected length of the regularity constraints defined in the initial step. Also, it indicates how well the corresponding edges of the 2 boxes align and also performs a size comparison. SmartBoxes can be compounded to form bigger blocks by simply adding the single objective functions of their included boxes.

The algorithm [146] provides real time processing by handling columns as they are scanned, as in a line sweep algorithm. Again it is suited to the case of proper lattices and it is based mainly on disparities. They first start by estimating the major planes. To do so, they fit minor planes at each point in the scanline (column). They use PCA [12] to determine the normals to these minor planes and subsequently derive the ground normal. Facade normals are updated as scanning progresses. If the angle between 2 subsequent normals is within a certain threshold, then the newer is preserved else a corner is recorded. Also, a test is done to ensure that the facade has not been translated forward or backward. The authors define a column function calculated on the consecutive angles at each point in the scan line. The function is interpolated to account for the difference in sampling densities between near and far points to the laser beams source. They map the resulting signal to the frequency domain via the Discrete Cosine Transform (DCT) and use Fourier analysis to estimate the dominant period as the secondary spike. The more adjacent scanlines added, the more accurate the period estimate becomes. Because, it becomes estimated from the secondary spike in the sum of frequencies of

multiple scanlines. Based on this period, the column function is then approximated as a square wave function, to allow a more regular determination of the sub-structure center and height.

In [147], they perform a 2-phase repetition discovery namely sparse and dense detection. Sparse detection comprises matching of upright SIFT features in the rectified image, from which they establish histograms of possible translations. Local maxima correspond to the approximate repetition information that suffice to represent most data translations. They only detect repetitions in the x-direction and filter out intervals of less than 30 pixels. They go on to evaluate the tentative intervals for each image patch, based on their defined repetition quality measure. The measure is based on the distance between a certain subset of SIFT descriptors contained in the patches linked by the examined interval. They claim the measure is powerful enough to favour the smallest valid repetition interval in case its multiples exist, penalizes noise and intervals resulting from high frequency regions. To handle the case of first and last of a sequence of repeated structures, they use the vector and its opposite in the quality evaluation. The output of this phase is a quality map for the interval of the same size as the original image. Dense detection is mainly about the localization of the vertical boundaries of the repeating units. They mark them where the values of the quality measure drops to a certain level across horizontal scanlines. The vertical boundaries are further enhanced by decomposing them based on a continuity score. It measures how well the repetition stabilizes over a range of 4 times the repetition interval. They look for local minima in the signal of the continuity measure to suggest a separating point for 2 different repetitive patterns.

Despite exploiting the distinctive cue of repetitive patterns, it is clear that algorithms in this category are cumbersome and heavily reliant on thresholds and heuristics.

3.5 Regularized optimization functions

Wong et al. [148] minimize the cost function of placing bounding rectangles. The pool of pivotal points is sampled from a probability map and dimensions of the rectangles are obtained from a predefined range. The placements are penalized based on 2 aspects which are configuration- and

data-based. The configuration aspect is characterized by local interaction between the rectangles. It penalizes adjacency defined by vertical and horizontal displacements that will ultimately lead to overcrowding. As for the data term, it favours rectangles with high confidence measured by the encompassed probability values. It also targets maximizing 3 measures: size, intensity homogeneity and contrast to neighbouring rectangles. They use a structure-driven MCMC sampler [149] called multiple-birth-and-death (MBD) to optimize the rectangles based on initial placements and putative extensions of lattices and grids. However, their final results with regular patterns are obtained after applying either the low rank constraint of [150] or the over-simplified architectural guidelines of [57].

In [151], Dai et al. pose their problem as a matter of localizing vertical and horizontal split lines. The lines are encouraged to spread evenly across the image and to coincide with edges separating semantically different regions. The regions identification is done through a Random Forest algorithm applied on the set of single pixels. Clearly, this leads to split up of structures which do not strictly follow the alignment of the rest of the components. Another piece of work of limited applicability, due to its restriction to a single lattice, is [152]. Xiao et al. formulate the detection of generalized translational symmetry as a block matrix, whose parameters are alternately optimised via graphcut and dynamic programming. Dynamic programming is about sweeping through all states in the Markov process and examining all actions for each state to set an optimal solution. It is generally perceived as a slow procedure.

In [1], they build a factor graph of higher order cliques on the images, based on structural aspects more sophisticated than spatial proximity. However, their nodes are Bounding Boxes (BBs) of preliminary segmented regions with the pixel assignment done as a region-to-pixel mapping of the chosen label without the capability of fine tuning the results. Also, proved by their reported inadequacy in localizing segment borders, the hardwired specification of thresholds on aspects like alignment, size similarity and regular spacing, will fail with inaccuracies in the segments and subsequent BBs formation. The way they handle size variations and the subsequent reliability of relative location priors is unsatisfactory, given that they use vertical and horizontal distances in their absolute form. In addition, their algorithm does not incorporate appearance in determining edges between the BBs, as they rely

on purely geometrical properties.

Zhao et al. in [153] initially find a set of translation vectors using a mean shift clustering algorithm and refined with a graphcut run. The selected vectors are further used to guide the segmentation process. Obtaining preliminary segmentations as such is misguiding to a great extent. Most often, structures are evenly distributed around others- walls surrounding windows. In this case, single translation vector is good enough for both structures and can hardly be used as a discriminating factor between different semantic parts. More importantly, sometimes segments are self-mapped rather than transfered elsewhere, because pixels are surrounded by very similar neighbours such that short translations transfer them to good enough matches. A problem that can be aggravated by their algorithmic preference of shorter translations. In their refinement step of the graphcut output, they use a thresholded agglomerative clustering of pixels. It is evident results of this phase heavily rely on appearance heuristics, prone to being data rather than application dependent. Their final energy function is counterintuitive as it assigns a data cost that is directly proportional to the probability of the label. Also, their framework lacks the incorporation of past data structural priors.

In [154] facades are subdivided into blocks which are then represented as a rank-one matrix. Each block is concerned with a pair of structures (wall and non-wall) resulting in a 0-1 encoding scheme. The rank-one approximated matrix is obtained through an optimization process utilizing augmented Lagrangian multipliers. The cost of the optimization function is the number of inconsistencies between the input matrix based on appearance classifications and the output 0-1 matrix. The inconsistencies are measured based on l^0-norm converted to a convex surrogate l^1-norm to allow efficient optimization. It is done in an EM framework while alternating between minimizing the pixelwise classification errors and the inconsistencies. Appearance classifications are obtained through Random Forest operating on HOG [35] and textons [155] of image patches. The original partitioning into blocks is done recursively with the help horizontal and vertical split lines placed at rows and columns with a maximum margin from non-wall structures. It is highly reliant on the correctness of the initial classification and it produces grids of structures over unified background but does not identify to which architectural structure it belongs.

The work in [57] has been upgraded in [58]. While, maintaining the overall framework of 3 layers, they incorporate deformable part-based model detectors for object localization and perform max-margin learning of the CRF parameters in grid graphs. In addition, they refine the facade configuration through integer optimization. However, at the final stage of the pipeline they resort once again to the heuristic weak architectural principles for post processing such as, rejecting a balcony hypothesis if it is not topped with a window and accepting running balconies only on specific floors of the building.

In [112], Kozinski et al. specify a user-defined shape prior of grid form, in which they embed constraints of adjacency. They categorize the boundaries between structure pairs into 3 subtypes: straight, winding and irregular, and ones of containment in hierarchical form. Their final model is the one that achieves the minimal number of penalties over adjacency patterns, optimized through the Viterbi algorithm.

The algorithm in [156], iteratively accesses the image, searching for specific structures in each iteration. Starting from the basic assumption that all pixels are wall ones, it then tries to replace this labeling while imparting row-wise optimized local arrangements of predefined adjacencies of window/balcony, roof/sky/chimney, and door/shop. The optimization boils down to a local decision determining the state of the primitives via dynamic programming. The decision depends on initial localization of architectural elements sampled from the probability map. The posteriors are obtained from a multi-feature extended vector [157], which has bag-of-words descriptors over SIFT, textons, ternary patterns and self-similarity in a randomized set of rectangles.

Regions are all enclosed by rectangular bounding boxes. While, this is acceptable in our application [113], it prevents the algorithms from being extended to more generic scenes.

We would like to position our work in the taxonomy of related work. We present 2 algorithms. One that belongs to the paradigm of minimizing the energy formulated from appearance and structural priors designed by the user but learned automatically from the training data. It is clear that the current optimization functions in the field are not rich enough in terms of the structural priors they incorporate. This is a significant disadvantage especially because many sub-optimal discrete com-

binatorial optimization algorithms have been developed over the past decade that are able to handle more complex functions efficiently. The second algorithm explores a new paradigm to our application. It is generating annotated outputs based on a latent representation. The latent representation itself is data-driven and modulated by a probability distribution again learned from training data. Statistical learning is particularly suitable for computer vision.

Chapter 4

Probabilistic Graphical Models and Semantic Segmentation

Many computer vision problems can be solved as a matter of selecting the model with the maximum probability. In the multi-label case of image segmentation, the probability distributions of the configurations depend on the priors of the assigned classes and the likelihood of these classes given the characteristics of the observations. All this information is captured in a learning phase ahead of inference. This formulation is referred to as Bayesian estimation and is expressed in by the following posterior equation:

$$p(\theta \mid x) = \frac{p(x \mid \theta)\, p(\theta)}{\sum_{x} p(x \mid \theta)\, p(\theta)} \tag{4.1}$$

Where θ is the set of parameters of the model, x is the set of observations.

The aim is to find the distribution parameters θ based on Maximum A Posterior (MAP). The problem is often solved in its dual form, by minimizing an energy function - the total loss incurred by the current parameters. Equation 4.2 highlights the inverse correlation between the energy and the model posterior.

$$E(\theta \mid x) = -\log\left(p(\theta \mid x)\right) \tag{4.2}$$

4.1 Random Field Optimization

Random fields are a logical choice for formulating image segmentation problems. They consist of an undirected graph of nodes and edges. The nodes correspond to image primitives (pixels and superpixels) or even more meaningful entities (sub-parts, parts, objects). Whereas, the edges define the topological relationships between the different entities. A clique is an important notion that determines the size of the neighbourhood of the nodes and is directly proportional to the order of potential of the random field. Each entity is allowed a single label from a set. There are several techniques based on the maximum a posteriori (MAP) estimate that tries to find an optimized configuration of assignments of the nodes to the labels. Conditional Random Fields (CRF) hold the positivity and Markovian properties of Markov Random Fields (MRF) and utilizes the bayesian rule of inference for estimating posterior of a configuration. However, they add one aspect to its formulation, which is that the posterior calculation becomes conditional on the data.

The markovian property is inherently local as it makes the labeling of an entity bound to its neighbours in the clique. However, the global optimality emerges from the fact that these cliques are inter-connected allowing the propagation of label choice to distant parts of the images. A commonly incorporated prior in the MAP is the smoothness prior which penalizes the assignment of neighbouring nodes to different classes. This is especially problematic for boundary regions where the right choice is to label the entities differently. A common work around is weighting the in-between edge with a contrast measure. Nowadays, we find CRF/MRF in post-processing phases of algorithms, as a handy general purpose tool to enforce spatial coherence and remove noise, even at the expense of boundary correctness due to the over-smoothing effect. There are several approaches to optimizing the involved energy function, which takes the following form:

$$E(f) = \sum_{p} D(f_p) + \sum_{p,q \epsilon \mathbf{N}} V(f_p, f_q) \tag{4.3}$$

The energy expresses the unary penalty $D(f_p)$ of assigning a certain label f_p to the visual entity p, which is normally based on the classification score of appearance feature vectors. Binary

penalties $V(f_p, f_q)$ encode the neighbourhood $\mathbf{N}$ regularizers. This interpretation of energy is the basis of most work implementing the framework for semantic segmentation [33]. In [18], Levin and Weiss propose a CRF formulation that combines top-down and bottom-up cues. The low-level image features are embedded in the pairwise term which is weighted by RGB color difference encourage labeling discontinuities to be aligned with region boundaries. Higher-level prediction is obtained via the dataterm. Based on displaced image fragment BBs, the algorithm penalizes assignment depending on the deviation between the pixel value and the mapped to value. The displacement is dependent upon a thresholded correlation score between the fragment and the translated to position. During the learning process, the fragments and the weighting are determined such that the log likelihood of the CRF is maximized. The CRF is subjected to a first order approximation to eliminate the need for inference step calculations with the utilization of each fragment and weight suggestion. The pool of fragments is initialized randomly. Selecting the fragments and correlating them with images at test time is similar in principle to the DPM, which fires upon coinciding the learned parts with the image.

There is a genre of algorithms that summarizes the location of a labeled region as a single scalar in a feature vector that is fed to a classifier. In contrast, CRF/MRF frameworks model the pairwise spatial relations of adjacency using an underlying graph structure of image primitives. The are several approaches that are used to solve RFs. The differentiating aspects between them is the connectivity pattern of the underlying graph, the size of the allowed pool of labels, and the order of the model-defined as the maximum count of variables in the same clique. We review 2 methods that are utilized in our contributions.

4.1.1 Graphcut Algorithm

Graphcut algorithm [121, 123] is one technique used to find MAP on CRFs. It is derived from the max-flow/min-cut algorithm of graphs[158], where there is a sink and a source node. A s-t cut is defined as the cut that disconnects the source from the sink in separate sub-graphs and leading to the association of each node to either sub-graph. The s-t cut selects the edges with the lowest possible cost, thus minimizing the energy. This formulation has lead to providing exact solutions to the binary

classification problems [122]. To accommodate the more common case where there are multiple labels, good approximation variants have been proposed which depend on selecting alternate labeling that greedily reduce energy, until no further reduction. This is referred to as move making techniques. However, exact global optimality is no longer guaranteed. α-expansion is one variant that outperforms another, namely, $\alpha - \beta$ swap [123]. In α-expansion, at each iteration, each label consensus set is expanded to include more nodes, and the algorithm favours the label whose expansion yields the maximum drop in the total energy. As for $\alpha - \beta$ swap, which exchanges the labeling between 2 nodes originally tagged with α and β at each iteration. There are submodularity constraints on the handled energy function by graphcuts. The pairwise potential must follow some properties known as Submodularity Constraints, that are given by:

1. $V(f_p, f_q) = 0 \Leftrightarrow p = q$
2. $V(f_p, f_q) = V(f_q, f_p) \geq 0$
3. $V(f_p, f_q) \leq V(f_p, f_r) + V(f_r, f_q)$

$\forall p, q, r \in$ set of entities to label and f_p, f_q, f_r are their putative labels, respectively.

Conditions 1, 2 and 3 define a metric function and are efficiently handled by α-expansion, while 2 and 3 define a semi-metric function solved by $\alpha - \beta$ swap. POTTS is invariably encountered when modeling the pairwise term in this algorithm as it satisfied the preceding conditions. It is defined as follows:

$$\delta(f_p, f_q) = \begin{cases} 0 & \Leftrightarrow f_p = f_q \\ 1 & \Leftrightarrow f_p \neq f_q \end{cases} \tag{4.4}$$

Based on the GrabCut [159] (a graphcut framework for image segmentation), Goring et al. [160] compare between 2 approaches. In the first, they propose obtaining preliminary objects segmentation based on MLE given Gaussian Mixture Models learned from the foreground and background pixels in the training sets. GMMs are further used to categorize the obtained foreground suggestions based on the color distance to the nearest neighbour, into various object classes. In addition, the authors use a shape prior in the form of 7 Hu invariant moments. In the other approach, they apply the algorithm

of Felzenszwalb [5] which results in a BB, which is then refined using Grabcut. The algorithm uses HOG features and a set of object parts.

Edges intrinsically define pairwise relations. This is the reason graphcuts primarily model first and second order potentials. There have been efforts to modify the formulation to include Higher Order Cliques (HOC) that involve more than 2 variables. One approach is adding auxiliary nodes that act as factors in factor graphs. In [161], they apply various reduction techniques to convert the multilinear polynomial resulting from the series of involved variables into the quadratic form that can be solved efficiently. Their proposed reduction procedure is able to convert group of higher-order terms at once. They compare it to other techniques such as reduction by substitution, reducing negative-coefficient terms, and reducing positive-coefficient terms methods.

One major issue when optimizing energy functions is that accuracy of the outcome is sensitive to the weighting of different terms in it. There are several efforts in this respect when using graphcut algorithms. In [162], Peng and Veksler design a segmentation criterion that is used to select the best set of parameters for the energy function in a binary classification setting of foreground/background separation. They formulate the goodness of segmentation as a binary classification. For this task, they train an AdaBoost classifier [163] using manually specified segmentation samples into good and bad. The samples are obtained via running the graphcut algorithm on the images with various parameter settings. [164] on the other hand, use a similar idea but positive samples are manually segmented samples versus randomized grouping of superpixels as negative samples. In [162], each segmentation is characterized by several features of: histogram-based intensity variations intra- and inter- region, consistency of the gradient direction, the number of corner on the boundary as an indication of its smoothness and textural features based on Gabor filters. The features are normalized based on a ranking scheme of each feature value with respect to its range and the absolute value is discarded. Another approach is presented in [165] for refined parameter selection when applying graphcut for localizing cells in microscopic images. The authors carry out a series of logarithmic image enhancements and morphological operations to extract cell boundaries. They go on constructing a special formula for calculating the multiplier of the smoothness term that is a subject in the intensity of

boundary and non-boundary pixels. The approach boils down to a different scheme for weighing the edges from the commonly utilized intensity contrast formula. Another example for region-dependent parameter estimation is seen in extracting regions of lungs from chest x-rays [166]. Local features of Haar-based texture and Hessian shape information are used to predict the parameter class based on a trained boosting classifier [167]. The parameter class is initially randomized, then refined through calculating the segmentation error resulting from the assigned label.

4.1.2 Sequential Tree ReWeighted (TRW-S) message passing

It is widely accepted that when the problem is in need for non-POTTS modeling and non-submodular functions, then methods such as QPBO [168], ILP solvers [169] and TRW-S [23] become the obvious choices. TRW-S is an extension of Belief Propagation for graph-based optimization. However, it is applied on a decomposition of the graph into a bundle of trees structures (cycle free graphs) which allows exact inference. The choice of the designated labels depend on max-marginal calculation on the sub-trees to which the nodes belong. Kolmogrov [23] enhances the formulation of TRW presented in [170] such that obtaining the global maximum on the lower boundary of the energy is guaranteed. This ensures yielding at least a sub-optimal solution for the optimization. TRW-S belongs to a class of algorithms called message passing. Message passing is primarily a reparameterization of the nodes and edges of the underlying graph according to some order scheme. Kolmogrov explains that the use of sequential order in message passing was found to be superior to the parallel one of [170] in accuracy terms. The algorithms also show efficiency gains as it allows the reuse of messages passed in preceding iterations in the forward direction from leaves to intermediate nodes. In [23], they define a Weak Tree Agreement (WTF) criterion, which is a state where the trees share a common MAP configuration. This entail that later iterations will not change parameters. They prove that their use of WTF as the stopping condition causes the sequential parameter update to converge to a local maximum.

In an application of learning distinctive visual attributes from images [171], TRW-S algorithm is used in a an Expectation-Maximization framework, in which both the latent features and the weighting

are simultaneously learned. Weights are estimated through a linear SVM, when the latent variables are initialized randomly. In its turn, the TRW-S is used to infer the latent values when the weights are fixated. The latent variables in this application are the attributes which are the regions of images that highly discriminate between different image categories and marked with bounding boxes. In the CRF formulation, the nodes are the images and terminal nodes rae the putative attributes. The datacost encoded in the unary term is the accuracy by which a classifier trained on the proposed attribute predicts the correct category of the image. As for the binary term, it encodes the similarity between pairs of images. And, it discourages the choice of regions that involve large spatial overlap to reduce redundancy. After the attributes are learned, they are used in fine-grained localization of similar structures in unseen images.

Despite the fact that TRW-S can be applied to the same potential functions as α-expansion, the latter is usually preferred in 2d multi-label segmentation applications. Because, POTTs model is utilized in the pairwise potential, which is handled most efficiently by α-expansion [172]. However, we find TRW-S more frequent in enforcing 3d shape priors [11, 173].

4.2 Restricted Boltzmann Machine

RBM is a variant of Hopfield Neural Network (HNN). HNN is an undirected graph-based algorithm that is used to find a latent (hidden) feature representation $\mathbf{h}$ of real-life observed data $\mathbf{v}$, by maximizing the joint probability $P(\mathbf{v}, \mathbf{h})$. Latent variables $\mathbf{h}$ exhibit complete linkage to the visible data nodes $\mathbf{v}$ and are introduced into the machine to mine for complex dependencies between the visible data. This is done to map the problem into a more expressive feature space for pattern recognition applications. Due to the complete linkage, RBMs are naturally fit for modeling HOCs. RBM is a generative model that operates by increasing the resemblance between a phenomenon and the model perception of the phenomenon. The energy to be minimized by the machine is of the following form:

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{b}^T\mathbf{v} - \mathbf{c}^T\mathbf{h} - \mathbf{h}^T\mathbf{W}\mathbf{v} \tag{4.5}$$

$\mathbf{W}$ is the weights on the undirected connections, $\mathbf{b}$ and $\mathbf{c}$ are the biases for visible and hidden nodes, respectively. In the MAP formulation, a well established approximation for the direction of search of the parameters is the gradient of the negative log-likelihood of the probability distribution function (equation 4.6). Optimally, the gradient should reach a zero value. In practice this is achievable within tolerance. Thus, learning proceeds in the direction of gradient descent.

$$\frac{\partial \log p\left(\mathbf{v}\right)}{\partial W_{ij}} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model} \tag{4.6}$$

Gibbs sampling is an approach that infers θ while minimizing the energy by calculating the expectations from the marginal distribution $P\left(\mathbf{v}\right)$ and $P\left(\mathbf{h}\right)$ as an approximation of the joint distribution $P\left(\mathbf{v}, \mathbf{h}\right)$. The approach is based on Markov chains Monte Carlo (MCMC), where the chain is defined as a compounded cycle comprising of a data-driven phase $\langle v_i h_j \rangle_{data}$ and a model-driven phase $\langle v_i h_j \rangle_{model}$. The alternation between the 2 marginal distributions continues until the parameters are refined enough so that the distribution stabilizes. Hence, the system has reached equilibrium (convergence). Gibbs sampling is stochastic in the sense that it relies on randomness in determining the states of the latent representation. The randomization was found to have a positive impact on the generalization ability of the model. Gibbs sampling assumes conditional independence among variables of a certain subset. This is a valid assumption when it comes to RBM as the restriction evolves from the inhibition of inter-connectivity within clusters of $\mathbf{h}$ and $\mathbf{v}$ nodes.

Contrastive Divergence (CD-k) [174] is an algorithm that transformed Gibbs sampling from a mathematical notion to an implementable procedure that alternates between deducing the probability states of the variables on the $\mathbf{h}$ and $\mathbf{v}$ nodes. The expectation calculation is attained by averaging over mini-batches that are used for gradient update. The procedure becomes tractable and follows reasonable complexity constraints. This is achieved by relaxing the requirement of running the sampling procedure till convergence, replacing this stopping condition with a prefixed number of sampling cycles. Experimentally, it has been found that even at a single cycle k = 1, good approximations of the distribution are obtained. It is the phase at which the first and second terms of equation 4.6 are

as close as possible, thus signaling a coherence between real-life data and its model-based reconstructions. Hence, the model perception of the phenomena is good enough. Also, visible data does not need to be sampled as in the standard Gibbs procedure, as they are pre-known from the realistic instances from the training data which ensures that the selected parametrized distribution is close enough to the asymptotic true one. This alleviates the need for determining the expectation from all the possible configurations of the observed data (as indicated in the first term in equation 4.6) and binds the learning to a finite set of training instances.

RBMs are often found as stand-alone learning layers, or stacked on top of each other to obtain higher level features with each RBM independently trained in a DBN. DBM is another variant with multi-layers, in which the update rule is dependent upon the visible, in addition to the hidden nodes of the abover layer. RBMs allow different modes of learning supervised, unsupervised, and semi-supervised, depending on the amount of unknown variables. The application of RBMs to semantic segmentation has been very limited. However, in the following we include other vision applications to highlight the capabilities of the RBMs and its variants.

The Shape Boltzmann Machine (SBM) of [175] is based on the Deep Boltzmann Machine DBM described in [176], in which the first hidden layer receives input from both, the visible and the top-most hidden layers. The image is subdivided into 4 overlapping tiles, each connected to a different subset of hidden nodes, which are forced to share weights. They restrict the number of hidden nodes compared to peer machines [177]. These tactics drop the number of free parameters to learn effectively and overcome overfitting on small datasets. Their first formulation handles single object segmentation in binary pixel representation. Later, they extend the work to allow for multi-part objects (MSBM) [178], in which the visible layer is replaced with multinomial nodes. The parameters of the appearance model are separately learned by Gaussian Mixture Models. In [179], the authors extend the SBM and MSBM to handle the case where seed pixels are provided for object sub-parts. They lay assumptions concerning the likelihood of pixels belonging to certain sub-parts to be inversely proportional to its distance to the sub-part seed. Thus, they alleviate the need for the labour intensive process of fully annotating the images. In the algorithm proposed by Kae et al. [180], the

region labeling decision is made by combining the potentials resulting from a Conditional Random Field (CRF) and RBM, using mean-field inference. CRF, acts as a regularizer for local consistency, whereas DBM is used as a global shape prior. It works with a superpixel representation of the image and has a virtual pooling layer that maps the superpixels to a grid-like input for the DBM. In a road detection application, Mnih and Hinton [13] make use of approximate localization based on road center lines to establish ground truth labels. They scale down the aerial image by applying PCA. Their algorithm handles patches of the images separately. Based on these patches, the RBM establishes the joint probability between the patch extracted features and the true road maps. At test time, the RBM is presented with the PCA features and, the resulting predictions are then refined through a neural network that is already trained to map between erroneous predictions and groundtruth maps.

In [181], they solve the limitation of SBM which is that subdividing the image blindly may lead to the breakdown of meaningful sub-parts of an object. The limitation makes the binary distributions of the patches inconsistent and harder to learn. The only difference is, in their work the patches are replaced with approximately convex shape polytopes obtained through a disjunctive normal shape model (DNSM) [181]. Each polytope represents a meaningful shape sub-part. The number and identity of sub-parts is pre-allocated to the RBMs of the first layer. Due to the static nature of training in the DBM, it is obvious that the machine will be able to handle one type of shape. For this reason, they only use it to complete silhouettes of people in pre-categorized walking and standing positions. In a weakly supervised setting, Heess et al. [182] separate foreground important objects from background clutter. They model both appearance and shape of the foreground objects by utilizing 2 sets of specialized visible nodes. First set consists of binary nodes for the shape mask. The other one models normalized pixel grayscale values. For the continuous values, they opt for a Beta RBM [183] as it is able to model variable statistical aspects of the visible nodes in contrast to a Gaussian RBM which deals only with fixed variance. They train foreground and background models separately using persistent contrastive divergence (PCD) [184]. The foreground regions are originally distinguished as outliers to the background model. Over the course of training the algorithm is able to detect persistent features of the foreground and are hence properly modeled.

In [185], the main aim was designing deep neural architecture that more closely resembles the human brain pathways of vision, while simultaneously improving its performance on vision tasks. They alter RBM configuration such that each hidden node is connected to only a subset of visible nodes comprising its receptive field. The key contribution is the adaptive determination of the receptive field of each hidden node during training. This results in non-uniform distribution of kernels over the image space with higher density in areas of high variability and vice versa. A similar feature can be found in biological circuitry. Gaussian-weighted masks are added as regularizers in the update equation the of weights, thus controlling the intensity of the hidden-visible connectivity. The receptive fields are allowed to move their positions at the end of each training epoch by reassigning its center to the strongest connection (based on weight) within its effective scope. The algorithm is tested in the domain of image completion for face figures.

Based on Max-Margin learning [186], [187] perform a 2-step algorithm in which a pretraining step is conducted using a standard SBM to learn object shape masks. In a subsequent step, direct connections are introduced between the raw image data and the visible and hidden layers of the trained machine. The motive is to jointly learn appearance and shape of objects such as pedestrians and animals. This necessitated a modification to the definition of the energy function and the activations of the layer variables. To learn the weight and bias parameters of the newly formulated machine, the authors carry out a concave-convex procedure[188]. In [189], the SBM is incorporated in a depth segmentation framework to resolve the problem of partially occluded object instances. The output is expected to be complete elucidation of the instances expressed in multi-layers. The authors use a NMS [190] energy formulation which is modified to accommodate a shape penalty that results from a SBM in probabilistic terms instead of binary as the original formulation of NMS.

An attempt to combine convolutional filters and RBMs is presented in [191] in which the goal is to learn a reliable feature space representation of images by adaptively learning filters to reconstruct images. It is done in a convolutional generative framework while imposing sparse connections inbetween layers to prevent identity mapping. A similar idea can be found in [192]. However, in [191], the feature maps are implicitly obtained through optimizing the cost function of the reconstruction pro-

cess. The function is optimized in an expectation minimization framework, while alternating between fixing the filters at one time and the features at another.

Wu et al. [193] use an integrated frame of CNN and DBN to achieve gesture recognition and segmentation in video streams. The 2 machines operate in parallel on different aspects of the input. On the one hand, a DBN works on skeleton dynamics expressed in the form 3D positional pairwise displacements of body joints with temporal variations. The DBN is trained in the 2-phase standard procedure: an unsupervised pre-training phase, followed by a supervised one relying on back propagation. Then, it is used to infer the gesture class on the top-most softmax layer. On the other hand, CNN takes as input normalized depth information and cropped color frames with zoom on specific joints. The authors mentioned that when comparing the outputs of the 2 tracks, the skeletal dynamics had a lower error rate. And, when combined they provide a boost to the accuracy. They tried 2 modes for integration. A simple weighted sum of posteriors outperformed the other which was concatenating the learned features at top layers of both machines and incorporating them in a separate learning and inference framework.

Chapter 5

Geometric Multi-model fitting

5.1 Introduction

Simultaneous parametric estimation of multiple primitive geometric models plays a key role in the interpretation of complex 3d scenes. This is characterized in the literature as a LP3 [194] problem, Irregular sites with discrete labels, on which techniques of unsupervised classification and optimization can be applied. The sites in our application domain are commonly data points contaminated by outliers. An outlier is a data point that does not unite with others to form a meaningful entity that describes the underlying structure being investigated. Pseudo outliers are the ones not relevant to a certain model but are actually inliers to another, whereas gross outliers are true noise points.

5.2 A brief review: existing approaches

Figure 5.1 summarizes our categorization of the work done in this field. In the following pages we review the significant algorithms highlighting their strengths and weaknesses.

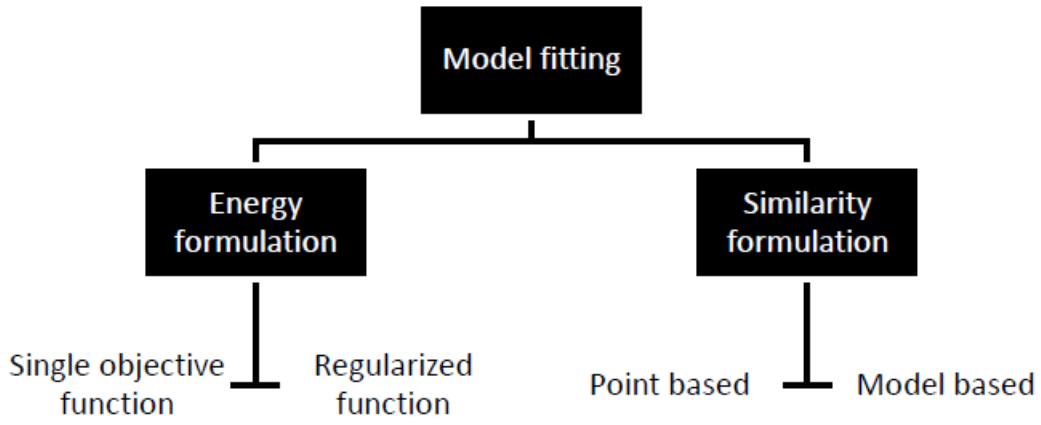


Figure 5.1: Categorization of work done in model fitting

5.2.1 Energy formulation

This section sweeps the literature, starting with the basic single objective energy function to the more regularized multi objective ones. Work belonging to this category fundamentally differ in two aspects, the cost function and the technique they apply to solve it.

Single objective function The discussion of model fitting must start with regression in the form of the Least Squares Fit (LSF) algorithm [195]. In this formulation, the cost function of establishing a model is only dependent upon the deviation of the data points from the model. Equation 5.1 shows the energy function minimized by the LSF. Given a set of data points p,

$$E(L) = \sum_{p} \|p - L\| \tag{5.1}$$

Where,

$$\|p - L\| = \text{dist}^2(p, L) \, \forall p \tag{5.2}$$

In other words, LSF estimates the parameters of the model, by minimizing value of the sum of squares of errors resulting from assigning the points to the model. This is done by solving a set of standard mathematical equations. Another important addition to the field is Random Consensus (RANSAC) [54]. It introduces a slight change in which the datacost is handled, as shown in the following equation:

$$\|p - L\| = \begin{cases} 0 & \text{if } \mathrm{dist}\,(p, L) < T \\ 1 & \text{otherwise} \end{cases} \tag{5.3}$$

Another popular variation is the MSAC [196], which again calculates the total datacost as the sum of squares of residuals (equation 5.4) as the LSF. However, it is the use of the threshold criterion that discards part of the points and thus provides a more robust algorithm against outliers.

$$\|p - L\| = \begin{cases} \mathrm{dist}^2\,(p, L) & \text{if } \mathrm{dist}^2\,(p, L) < T \\ T & \text{otherwise} \end{cases} \tag{5.4}$$

RANSAC and its variations tackle the problem of energy minimization with the use of greedy heuristics. It starts with randomly populating a set of models with the minimal set of sampling points. These models compete in having the maximum sized set of inliers and the winning model is optimized using its determined consensus set. The main weakness of the RANSAC lies in its reliance on the cardinality of inliers set assumption. It fails as a discriminating criterion between cross structures and true structures in case of multiple models in presence of gross outliers. Nevertheless, RANSAC has provided the basis for the expand/re-estimate framework used even in the most recent advances of the field, the PEARL [197] method, explained shortly. In addition, its way in forming the initial set is found in nearly all the systems that followed it.

The RANSAC adaptation to the multi model applications appear in its sequential version [198]. In each iteration, a single structure is optimized and its consensus set is removed from the field for a different model to be found in the following round.

The MLESAC is another RANSAC variation, presented in [199] and based on minimizing the

error in its negative log likelihood form. It is originally tailored for fundamental matrix estimation between two views. Firstly, Expectation Maximization (EM optimization) is applied to estimate the mixing parameter for determining the probabilities of inliers and outliers, utilizing an initial set of putative models. Then, the sampling set of correspondences that minimizes the overall energy is passed to the second stage of gradient descent method, to optimize the generated model. The obvious shortcoming of their work is the limited efficiency of the EM approach being susceptible to local minima.

A more recent study of a unified framework for RANSAC-variants is presented in [200]. The authors report enhancements in the 2 main stages, the hypothesis and verification of models. In the initial finding of minimal sets in the hypothesis phase, a grouping of data points is suggested which guides the sampling instead of the random procedure. It is either based on the spatial proximity, the similarity often seen when finding correspondences or another domain specific criterion. In the verification step, the effort has been mainly directed towards reducing the computational time by early discard of unpromising models subsets of pilot data points. Technically, these additions to the algorithm improve its overall performance. Besides our skepticism of the utilized techniques for the guided sampling, we have comments on more fundamental issues with the framework. Their definition of an *interesting* model as having an all-inlier minimal sample set does not signal a good model. Sometimes, an outlier point contributes better to the formation of a model, if it guides the model in a direction that causes it to better spread among its inliers. The reported model degeneracy criterion as an indication of model goodness is focused on the feasibility of producing a unique solution based on a given minimal set. It is a sort of constraining the models using application specific knowledge. However, for the accepted models, the algorithm preserves the basic criterion of goodness as being the number of supporting points. A criterion that we regard as inherently biased against underlying with consensus sets of low cardinality. In high levels of contamination, incorrect models can gather supporting data points surpassing that of real models. We view the consensus power as a matter of density rather than count. Also, we regard the independence of the hypothesis and verification phases as a limiting factor for the quality of the generated models. Even if the models

are refined later, it is carried out only based on local consensus sets. We would like to see in the RANSAC-based framework, progressive hypothesizing of models in the course of the algorithm, such that the selection is not bound to initial partially informed set of hypotheses. There is another problem that is more related to our specific algorithm and will be discussed in more details in the following chapter. If we opt for clustering of models, then RANSAC will not guarantee the existence of multiple variants of the optimal models which will then be prone to be dismissed as noise.

The oversimplified single objective formulation works by enhancing each model locally. It deals with the deviations of the points from the model independently while overlooking the spatial interrelations between the data points. This leads to failure to take into account cues that are inherent to the human vision system. These include, the density of points in areas that belong to the same model and the intuitive merging of adequately similar models. This gave rise to the need for a more principled approach.

Regularized function This is a realization of labeling within global contextual constraints, in which a number of opposing forces dominate the scene. The work presented in this area is generalized to a wide range of problems unless otherwise stated. Commonly, the problems include estimating affine models, homographies and fundamental matrix estimation, and motion segmentation.

Traditionally, the problem has been mapped on the uncapacitated facility location problem (FLP) [201] of the operations research field. For this reason, there are a couple of attempts that rely on "information criterion" AIC [202] derived formulations. The function incorporates the total transportation cost and the establishing of a new shop cost incurred in the FLP problem. The variation is in the techniques of solving the function. These include reversible jump simulated annealing [203], branch and bound [204], linear programming [205] and EM [206].

Perhaps the most comprehensive formulation is the one presented in PEARL [197]. It operates under the principles of Markov Random Fields (MRF), in which a neighbourhood system between sites is established and their joint probabilities appear in the energy function. As shown in equation 5.5, they added a smoothness prior that incorporates the spatial coherence in the search. It is expressed

as an energy term measuring the extent to which the smoothness assumption is violated by the selected configuration. Their postprocessing is an adhoc merging and splitting process.

$$E(L) = \sum_{p} \|p - L_p\| + \alpha \cdot \sum_{(p,q)\in\aleph} w_{pq} \cdot \delta\left(L_p \neq L_q\right) + \beta \cdot |\mathcal{L}_{\mathcal{L}}| \tag{5.5}$$

The algorithm runs in an iterative manner. The random initial set of hypotheses are verified using the α -expansion graph cut optimization. The implicitly assigned points are used to re-estimate model parameters and the feedback is once more passed to the α -expansion until convergence.

Recently, Yu et al. [207] presented their novel energy function. In essence, they are taking into consideration the same factors as in PEARL [197]. However, they differ in the means the terms are computed. Our take on their approach is the over emphasis on the smoothness assumption, evident in the inlier similarity and embedded in the model fidelity terms. Because, inliers of a model are not solely determined based on residuals but rather on the presence of similar points in the consensus set of a model. Theoretically, this could be amended by the proper assignment of multipliers for these terms. A more serious problem is the inlier similarity. This is not based on spatial proximity but on their residuals towards various models. This can be very misleading in case of random generation of models that may result in cross structures (figure 5.2). However, their redundancy eliminating term (regularizer) is more physically meaningful than the label cost terms in the formulations that proceeded them. Besides, they pose their energy in the standard Quadratic Program (QP) form handled by efficient constrained optimization techniques. This step produces a ranked list of probable structures preferable over the early hard assignment of points to models.

Comparative analysis of the regularized energy approach is a non-trivial task. One needs to examine aspects like the correctness of the energy function, by which we mean the existence of an optimal/sub-optimal solution as a function minimum. In addition, the efficiency of the optimization algorithm and its suitability to the energy function are to a great extent dataset dependent. Generally, the main shortcomings of this paradigm are the determination of the trade-off between the various energy terms and settling for approximate solutions to preserve computational feasibility.

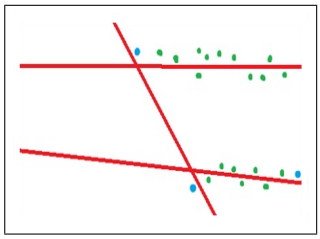


Figure 5.2: A snapshot of point arrangement showing 3 randomly formed models (lines). The points in blue share very similar preference based on the cross structure despite belonging in different models.

5.2.2 Similarity-formulation

This category exploits the fact that a structure can be detected by the presence of several entities sharing a certain property. The entities are the given points or some preliminary models. The property is usually defined upon a parameter, residual or conceptual space.

Point-based One such system is proposed in [208]. An agglomerative algorithm clusters points and the final models are the best fits of these clusters, based on the LSF. The points are expressed by their set of preferred models and a Jaccuard distance is defined as the linkage weight. The clusters are filtered by a size rejection criterion. Again, the preference is determined by thresholding, as in formation of consensus sets of RANSAC. In our opinion, their assumption "Residuals for each data point have peaks corresponding to the true models" is valid as long as the true model manifests itself with a considerable set of similar structures. This however cannot be guaranteed.

Model-based approaches that deal with the models directly, bypass the explicit individual assignment of points to structures. Theoretically, this results in a more globally accepted solution.

Hough transform is commonly used to group models based on their parameters. In [209], parameters are binned and modes are found directly from the histogram. Whereas, in [29] they are mapped directly to the parameter space, on which they applied the mean shit algorithm. There are well-known difficulties associated with mode finding, such as choosing the binning width and window size in the

mean shift algorithm. These affect the suppression and emphasis of local maxima thus accuracy. More importantly, one might argue that it is not straightforward to establish a perceptually uniform space based on the parameters of the Hough transform.

Zhang et al. [210], presented a system that operates in the residual space. In their work, they assume different modes in the histogram of residuals of each point, correspond to different real models. This is a fragile hypothesis in case of a high level of noise, in which a lot of points are equidistant to various models. A global decision is made with respect to the number of models, by taking the median of proposals of points. However, the final formation of the models is left up to the histogram of a single point, with an intervention of another point, in an unjustifiable way.

Chapter 6

Multi-model fitting based on Minimum Spanning Tree

6.1 Introduction

The problem of simultaneously establishing geometric models such as plane, homography and fundamental matrix fitting can be formally stated as, $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n}$ be a set of n data points. It is required to find $\boldsymbol{L} = \{L_i\}_{i=1}^{M}$, such that $\boldsymbol{L}$ is a set of models that best describe $\mathbf{X}$ and to assign each $\mathbf{x}_i$ to one of the models in $\boldsymbol{L}$. L_i is the parameter vector of model i which, together with the variable M are unknown a priori. In addition, the data points are contaminated by varying levels of outliers.

We propose an approach that relies on aggregating similar hypothesized models in clusters and electing the most probable from each cluster. The models are hypothesized via guided sampling based on Minimum Spanning Trees. We replace the commonly encountered criterion of goodness of models related to consensus set size to a more abstract one concerned with the stability the generated models. This acts as an indication of the conformity of the hypothesis to the underlying structure.

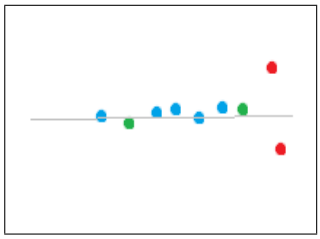


Figure 6.1: A snapshot of point arrangement showing 2 inliers in green with an in-between distance larger than the distances between one of them and the gross outliers in red.

6.2 Proposed Algorithm

A *good set* of initial hypotheses is vital when applying clustering techniques in multi-model geometric fitting. By a good set, we refer to the repeated presence of optimal/sub-optimal models, in order to form agglomerated dense regions in the model space, so that the correct models are not dismissed as outlier entities in the clustering procedure. Unfortunately, random sampling does not ensure the set of initial hypotheses will hold this property of being a "good set". In its best case scenario, it may provide the optimal/sub-optimal models in the initial set but not in high frequencies, as required. The mathematical foundation for determining the count of putative minimal subsets required to build the models and at the same time ensure the presence of the correct ones only guarantees one all-inlier hypothesis and it increases dramatically with the slightest increase in the level of noise in the data points.

In the following, we discuss the size aspect of the initial random set. p is the cardinality of the minimal sample set necessary to establish a model. For a single model i , ϵ_i is the ratio of inliers of the model to the total number of points. ϵ_i^p is the probability that all p points are inliers. And, the probability that at least one of the p points is an outlier is $1 - \epsilon_i^p$. Then, $\rho^{'}$, the probability of failing to form at least one all-inlier sample in m_i withdrawals, should be as expressed in equation 6.1:

$$\rho^{'} \leq (1 - \epsilon_i^p)^{m_i} \tag{6.1}$$

$$\log\left(\rho^{'}\right) \leq m_i \log\left(1-\epsilon_i^p\right) \tag{6.2}$$

$$m_i \geq \frac{\log\left(\rho^{'}\right)}{\log\left(1-\epsilon_i^p\right)} \tag{6.3}$$

In experiments, $\rho^{'}$ is usually fixed and set as low as 0.01 at maximum because model finding algorithms do not afford to work on an initial set that lacks the correct model. In the general case of having k models, a naiive solution to estimate the total number of samples is to add up the number required for each model $\sum_{i=1:k} m_i$. This will incur much redundancy because a sample that fails to be all-inlier to one model may fulfill this condition for another. Also, the inliers cardinality of each model is hardly known apriori. For this reason, an estimate of the cardinality of the smallest set of inliers should be used. This provides the highest lower bound on m.

As stated above, the number required in the initial set scales in case of multiple structures as a consequence of a decline in ϵ. This is because the added models introduce *pseudo* outliers over and above the *gross* outliers already present due to the imaging artifacts. This leads to shrinking the inliers to total number of points ratio thus increasing m. Bearing in mind that this only ensures one all-inlier sample per model, we can imagine the upsurge if many all-inlier samples are needed. For this reason and because our algorithm is dependent on the existence of multiple analogies of the optimal model, we have resolved to the guided sampling paradigm.

6.2.1 Minimum Spanning Tree (MST) guided sampling

We propose an algorithm that is generic for Euclidean image space and which belongs to the group of work that focuses on increasing the probability of hitting an all-inlier sample [211]. One approach to categorizing points into possible consensus sets of different structures is relying on spatial proximity, as in [212]. The principle implies that possible inliers are closer to each other than to outliers. But, as we have shown in figure 6.1, this produces errors in the presence of outliers. Also, it is a well known geometric fact that when building structures out of a consensus set, it is better to sample far apart

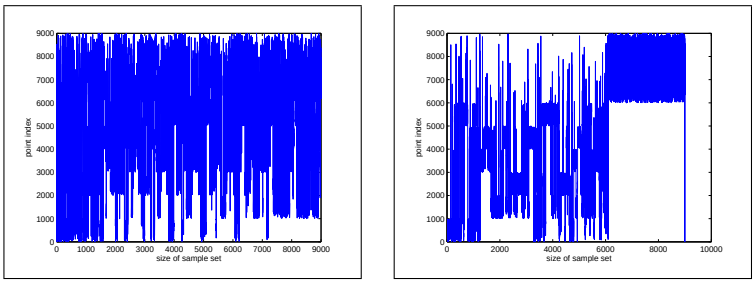
Input: $\mathbf{X}$: set of data points
Output: $\boldsymbol{L}$: set of found models
for *each* $\mathbf{x}_i \in \mathbf{X}$ **do**
  $T_i \leftarrow \phi$ */*T_i: subtree of the MST originating at $\mathbf{x}_i$*/*
  while $\mathsf{size}(T_i) \leq z$ **do**
    expand T_i by adding a point from the MST originating at $\mathbf{x}_i$
    find the current best fit model of T_i
    calculate model deviation (equation 6.6) based on current and preceding best fit models
  end
  localize valley_of_interest in smoothed model deviation curve as in figure 6.4 (b)
  find $T_i^{elected}$ which corresponds to the minimum margin of error as in figure 6.4 (b)
  construct model of best fit to $T_i^{elected}$ and add to initial set
end
construct the similarity matrix between models according to equation 6.8
perform repeated-2 spectral clustering of models based on the similarity matrix
find centroid models of the clusters and add to the final set of models $\boldsymbol{L}$

Algorithm 1: Proposed algorithm for model fitting

points to provide a better fit to the whole set of inliers. In addition, methods that sample locally based on proximity tend to produce isolated patches of models in case of gaps due to partial occlusion. While, sampling based on residual as in [211], is capable of joining disconnected patches of points of a model.

We go for a compromise between proximity and spread. We begin by deterministically forming the tentative models. At each point we initiate a sample set. Gradually, this set is expanded by incorporating more points. With the addition of each point, we find the best fitting model of the formed set, the absolute residuals $\mathbf{r}$ of the points in its sample set, and the number and absolute residuals $\mathbf{s}$ of the points in the consensus set of this model. The consensus set is the points in $\mathbf{X}$, which exhibit an acceptable residual to the model. The acceptable values lie in the interval $[0\,,\,\mathrm{mean}\,(\mathbf{r}) + \mathrm{std}\,(\mathbf{r})]$.

For the expansion process, seed fill algorithms are a common choice. However, we resort to the Prim's algorithm [213] for finding the MST, because in our applications, point clouds and feature points of sampled images, are represented as Euclidean Complete Graphs (ECG). Front propagation algorithms are ill-defined over ECGs, because there is a direct path between all-pairs. So, when the target is the shortest path, then it will be only one edge, which is the direct edge. Choosing a neighbourhood graph as a means of representation using thresholding of the distances or voxelization



(a) (b)

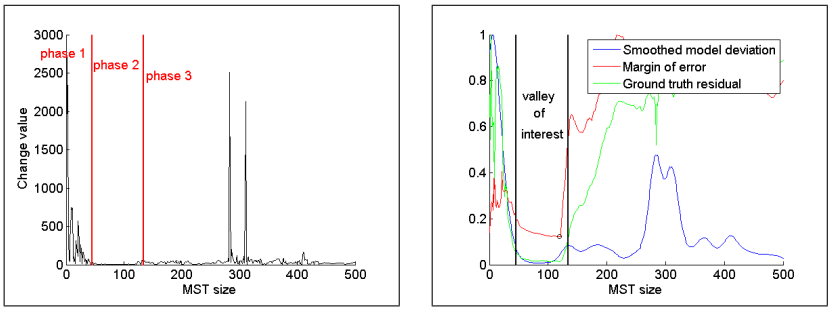
Figure 6.2: (a) Graphs showing the index of the point added at each iteration of the expansion of the sample set based on Front propagation; (b) based on Minimum spanning tree. In this example, points with indices > 6000 are noise points. It is evident that the bulk of noise points are accessed at late iterations after all the models points have been visited.

(a) (b)

Figure 6.3: (a) MST of size z initiated at some point. (b) Subset of the MST that satisfies our criteria.

is a possible solution but it has the undesirable effect of inducing multiple Connected Components (CC) and binding the formed models to the localities of these CCs. In our view, the terminating criteria for the spread of a model should be based solely on the decline of its fitness, to be able to account for signal interruptions of partial occlusions. More importantly, we argue, the MST is more robust to noise. Bearing in mind, the geodesic path between 2 inliers of a model is on its surface, the MST will start by spreading over this surface (figure 6.2). This happens because the MST algorithm is devised to extend to the points closest to the skeleton of points already formed. When most of the surface points are visited, the algorithm backtracks to visit other points in the vicinity of the surface. These have a higher possibility of being outliers. Hopefully, at this point the established model will be more fitting to its inliers, and will be less affected by the addition of outliers. Seed fill algorithms, on the other hand, blindly flood the nearest neighbours in a breadth first search manner. This adds points not closest to the skeleton as in the MST approach, but rather to the recently added data point. The problem occurs when this recently added point is itself an outlier. In this manner, it will act as a leakage point through which the expansion algorithm will proceed through other outliers. This is more prone to adding noise points earlier in the expansion as in figure 6.1. Even if a criterion more elaborate than the closest neighbour was used to direct the traversal of the graph, such as a gain in the model likelihood, its lack of a backtracking mechanism causes good candidate points to be lost at high levels of the traversal tree. To sum up, it is the backtracking advantage of the MST over seed fill algorithms that boots its ability as an expansion algorithm in aggregating more promising points in light of the established skeleton rather than the last point. This makes it a better candidate to be utilized by our algorithm. At each iteration of Prim's algorithm, the edge with the least value connected to the growing spanning tree is selected, provided that it increases the number of nodes in the tree.

The fundamental question is when to stop the growth of the MST. As we mentioned before, each point generates a set of plausible models. Each model corresponds to a subtree of the MST initiated at this point. The growth continues until a maximum size of z is reached and we select the optimal subtree by combining 2 criteria namely, *stable alignment* followed by *margin of error*. We start



(a) (b)

Figure 6.4: MS sub-T selection criteria (a) A model deviation signal showing a typical behavior of models constructed from subsets of the sample set; (b) A graph showing the smoothed model deviation, margin of error, ground truth residual. Two vertical lines marking the valley of interest. The circle shape marks the chosen subtree size for this point.

examining these generated models at each point, by recording the *model deviation*, indicating the degree of change that happened to the best fitted model over the previous single expansion step. The model deviation is basically a dissimilarity measure $d_{i(i-1)}$ between consecutive models. Typically, the graph of model deviation (figure 6.4 (a)) can be divided in 3 phases:

- *Phase 1*: It is characterized by sharp ripples. This shows the models undergo substantial changes at the start of the MST growth process. Because, the addition of a single point (inlier or outlier), makes a profound effect on the alignment of models of smaller sets of points.

- *Phase 2*: It can be described roughly as a plateau region. This indicates a stability in the model formation. The reason is, the growing size of the subtree enhances the spread of inlier points and the adherence of the generated model to the underlying structure. Thus, the inclusion of gross outliers in the subtree does not alter the alignment of the model. As long as, the density of the inliers exceeds that of the outliers in a manner that allows the presence of an underlying structure in the first place.

- *Phase 3*: The deviation value increases again, mainly, due to the inclusion of tangent pseudo outliers.

The model deviation should capture the change in alignment when viewed globally as accurately

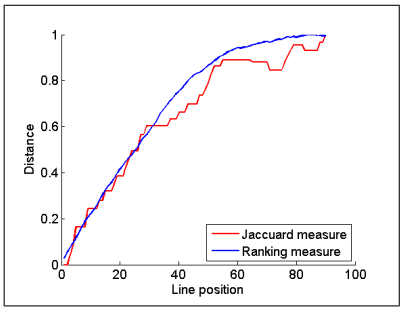


Figure 6.5: Normalized d_{0i} between the rotating line and the x-axis at each angle position.

as possible. Due to the irregular nature of the space on which the models are defined, we opt for an arbitrary dissimilarity measure. We quantify the difference between models L_i and $L_{(i-1)}$ by starting with an ordering of the n data points according to each model L_i. First, we calculate their absolute residuals to the model to form the residual vector:

$$\mathbf{r}^{(i)} = \left[r_1^i r_2^i \ldots\ldots r_n^i\right] \tag{6.4}$$

This vector is subjected to sorting in non-descending order. The new indices given to all points are recorded in $\mathbf{a}^{(i)}$:

$$\mathbf{a}^{(i)} = \left[a_1^i a_2^i \ldots\ldots a_n^i\right] \tag{6.5}$$

The dissimilarityd_{ij} is calculated as the total deviation in the sorting of points according to L_i and $L_{(i-1)}$, as follows:

$$d_{i(i-1)} = \sum_{o=1}^{n} \left| a_o^{(i)} - a_o^{(i-1)} \right| \tag{6.6}$$

This dissimilarity measure resembles the Jaccuard distance defined on preference sets in [208], but with a more global outlook and does not suffer from the deficits of residual thresholding. To illustrate how the measure operates, we sampled a number of 2d points around a line, and then the line, was incrementally rotated by an angle of 1° for 90 times. At each angle position, the d_{0i} was recorded based on both the Jaccuard and our ranking dissimilarity, (figure 6.5). Our measure was shown to be more linear and sensitive to small perturbations. In figure 6.4 (b), it is evident that phase

2 of the model deviation roughly coincides with the lowest values of the average residuals of the ground truth model points, suggesting a good fit of the generated models in this phase. We localize this region in the graph by first convolving the model deviation signal with a smoothing 1d Gaussian filter. We perform a 1d watershed transform and find the segmented part of the first basin whose local minimum does not coincide with the first index of the model deviation vector. We then perform thresholding to trim the heights of the peaks of the valley to the shortest of them. We refer to this procedure as applying the stable alignment criterion. To elect the best fitting model, we seek the sample subset with the least margin of error t_i defined as:

$$t_i = 1.96 \times \frac{\mathrm{std}\left(\mathbf{s}_i\right)}{\sqrt{\mid \mathbf{s}_i \mid}} \tag{6.7}$$

$\mid \mathbf{s}_i \mid$ is the size of the consensus set of the model i. We utilize t_i as the *goodness of fit* in our algorithm. The larger the margin of error, the less confident one could be that the data points resulted from a true structure. Geometrically, it indicates how well aligned and dense the points are in the consensus zone.

One of the challenges for this algorithm is, tangent models. One can argue that MSTs resulting from intersecting models constitute a small fraction in the first place, as explained in [214]. In addition, exploiting some domain specific knowledge to eliminate edge points will radically solve the problem. Nevertheless, we try to provide a generic algorithm that will act blindly on point clouds. Also, in our application, edge points are among the highly sampled genre of points, as they hold strong characteristics of a region. Figure 6.3, shows that our combined criterion is capable of choosing the subtree of the MST that truly belongs to a single model. The initial set is then formed of the best fits to the MST subsets. The RANSAC recommendation provides the ceiling for the size of the initial set. In practice, our algorithm managed with a fraction of this quantity. An advantage of our algorithm is, the whole set of points is available each time a new model is formed. In contrast to the multi-RANSAC, when the set of inliers of a model are completely eliminated in future iterations. In contrast to previous work on MSTs for clustering [215], we combine proximity with the stability criterion to promote it from a proximal point aggregation tool to one more capable of model detection.

6.2.2 Multiplicity guided model detection

True structures in our algorithm manifest themselves with multiple structures that are relatively close to them in the initial set. This multiplicity is a quality that reinforces the existence of an underlying structure or stimuli. To exploit this principle computationally, we have utilized the spectral clustering technique. It is specifically suitable for this application, because most other methods need a definition of the absolute location of the elements in some space. In contrast, spectral clustering relies on the similarities between the elements. Thus, we construct a difference matrix of size $m \times m$ in which each cell (i, j) indicates the degree of dissimilarity d_{ij}. It is then converted to a similarity matrix as follows:

$$w_{ij} = 1 - \frac{d_{ij}}{\max_{\forall i,j} d_{ij}} \tag{6.8}$$

Again, for the calculation of d_{ij} , we use equation 6.6 to assess a conceptual dissimilarity based on the models rankings of points. The similarity matrix is then passed to spectral clustering [216] to produce subsets. This paradigm of clustering is based on combinatorial optimization concerning graph partitioning [217]. It emerged from the observation that eigen vectors derived from eigen values of graph Laplacian hold a lot of information about the properties of graphs. It quickly gained appraise due to its empirical merits, the ease of utilization of standard linear algebra methods and scalability. The putative models are represented as nodes/vertices in an undirected weighted graph G, written as an ordered pair $G = (V, E)$, where V is the set of vertices and E is the set of edges. The vertices belonging to an edge are called the end points. The adjacency matrix A is a symmetric positive semi-definite holding the similarity coefficients between L_i and L_j. The degree of a vertex is defined as

$$\deg(v_i) = \sum_{j \epsilon V} w_{ij} \tag{6.9}$$

and the degree matrix D for G, is an $m \times m$ square matrix defined as

$$d_{ij} = \begin{cases} \deg(v_i), & if\ i = j \\ 0, & otherwise \end{cases} \tag{6.10}$$

A subset $X \in V$ is called a connected component, if there are no edges between X and $\overline{X}$ and X is connected (i.e. if any 2 vertices in X are joined by a path, then all intermediate points belong to X).

The algorithm is initiated by constructing the A matrix. The kernel normally used in this step is the gaussian one, defined by equation 6.11.

$$A_{ij} = \begin{cases} \exp\left(\frac{-\text{distance}(i,j)}{2\sigma^2}\right), & if\ i \neq j \\ 0, & otherwise \end{cases} \tag{6.11}$$

Alternatively, we can apply the following monotonically increasing transform function

$$A_{ij} = \begin{cases} 1 - \text{norm\_distance}(i,j), & if\ i \neq j \\ 0, & otherwise \end{cases} \tag{6.12}$$

In our work, we chose to implement the normalized graph Laplacian P [218] (equation 6.13) rather than the unnormalized version [217].

$$P = I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \tag{6.13}$$

The former corresponds to the N-cut of a graph, as opposed to the latter which corresponds to the min-cut. The eigen values and vectors are then computed. The smallest values and their corresponding vectors hold the essence of the graph properties. In general:

- If G is connected, first eigen value is 0 the vector is $\mathbf{1}$. Whereas, second eigen vector corresponds to the Fiedler vector. This gets the min N-cut indicated by the sign of its elements in the binary case. The corresponding eigen value gives the summation of the weights of edges cut.

The subsequent eigen vectors encode further partitionings of the formed subgraphs.

- If the graph comprised z connected components then the multiplicity of eigen value 0 is z.

In reality, the trivial task of finding the first 0 eigen values is complicated by existence of noise and the descriptive capacity of the similarity coefficients. The formation of the oriented incidence matrix Q is done by stacking the z eigen vectors with the smallest eigen values, as columns in a matrix $\in \mathbb{R}^{m \times z}$. The resulting row vectors $y_i \in \mathbb{R}^z$ correspond to the models that need clustering and the final decision is done using the *K*-means algorithm.

We handled the issue of the unknown number of models and subsequently clusters with the *Repeated 2-clustering method*. Clustering of similar models is done hierarchically in a top-down approach. Each node is hypothetically split into 2 subsets. The regularization function is the *Daives Bouldin* (DB) measure [135], used for internal evaluation of clustering and here it is calculated using all the current clusters. If this break down introduces an improvement i.e. decreases DB index, then it is carried out. Otherwise, the node is left as a leaf. At the end of examining all the nodes, the clusters are the leaves.

Daives Bouldin index (DBI) [135] is a measure for internal evaluation of the clustering used frequently for comparison purposes of various techniques. DBI is one of the best performing indices. This is justified in the comparative study of [219] and in our experiments. In figuring out the optimal number of partitions, DBI examines the ratio of the inter-cluster distance to the intra-cluster distance. In effect, it balances the diversity with coherence. Normally, the distances are calculated with respect to the centroids of the clusters. However, in our application, we calculate them for every pair of inter/intra-cluster points. This does not incur a computational overhead as the values are already present in the distance matrix. DBI is calculated as follows:

$$DBI = \frac{1}{b} \sum_{i=1}^{b} \max_{i \neq j} \left(\frac{\sigma_i + \sigma_j}{d\left(c_i, c_j\right)} \right) \tag{6.14}$$

Where,

b : the number of clusters c_x : the centroid of the cluster σ_x : the average distance of all elements

Method	Random sampling	Multi-GS	MST-GS
Example 1	0.3875	0.3418	0.3328
Example 2	154.4047	93.7360	12.3048
Example 3	41.5376	10.3648	1.8447
Example 4	0.0551	0.0340	0.0244

Table 6.1: Average per-model residual of closest group of ground truth model points to their best fitting models in the initial set.

in the cluster to its centroid

For selecting the final set of models from the clusters. We find the centroids $L_e^{(i)}$ i.e. model that is least dissimilar to the rest of models with it in cluster c_i, (equation 6.15). In contrast to RANSAC, this approach is unbiased to models with large consensus sets.

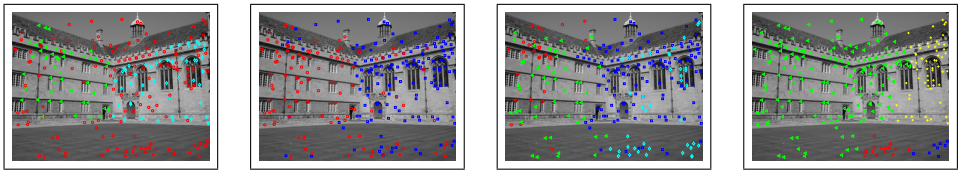
$$L_e^{(i)} = \arg_{L_k^{(i)} \in c_i} \min \left(\sum_{j=1}^{|c_i|} d_{kj} \right) \tag{6.15}$$

6.3 Experimental Evaluation

We validate our proposed algorithm by testing it in the applications of plane, homography fitting, and motion segmentation in the case of multiple structures. In our experiments on estimating homographies and motion segmentation, we assume the correspondence problem is solved by matching of SIFT descriptors [73] and preconditioned point matches are readily available. We follow [220] in the use of the Direct Linear Transformation (DLT) algorithm for fitting the homographies and the symmetric transfer error for calculating residuals. In case of motion segmentation, we use the normalized 8 point algorithm for estimating the fundamental matrix for each motion and the squared Sampson distance for the geometric errors. As for the planes application, Principle Component Analysis (PCA) is used to establish the fits and the residuals are the perpendicular distances from the points to the planes. We hereby present 4 examples [211], whose ground truth information is available in

	Recall			Precision			M		
	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)
Ex.1	0.9143	0.8552	0.9672	0.6094	0.7004	0.6448	9	9	6
Ex.2	0.7115	0.7692	0.8316	0.7253	0.4657	0.6014	4	4	2
Ex.3	0.9164	0.9443	0.9897	0.9298	0.9011	0.9404	5	5	3
Ex.4	0.9950	0.9077	0.9915	0.7803	0.6229	0.5271	5	5	3

Table 6.2: Average recall, precision values and the count of detected models for results of (a) J-linkage, (b) PEARL, (c) Our proposed algorithm .



(a) (b) (c) (d)

Figure 6.6: Wadham. (a) Ground truth (outlier points added shown as red circles). Results of (b) proposed algorithm; (c) J-linkage; (d) PEARL (results are different from those reported in their paper because utilized parameters of energy function were not given).

the form of membership of points.

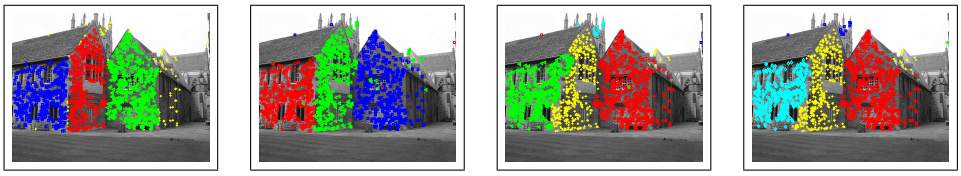
- *Example 1*- Synthetic cube of 6 planes. Each cube face consists of 1000 points, outliers count=3000 and Gaussian noise= 0.015.
- *Example 2*- Wadham is a homography example of two planar regions. Models consist of 52 and 65 points respectively, outliers count=176.
- *Example 3*- Merton III is a homography example of three models. Models consist of 498, 394 and 570 points, outliers count= 90.
- *Example 4*- Dino books is a motion segmentation example that has three motions. Models consist of 78, 86 and 41 points, outliers count =155.

6.3.1 MST guided sampling

In this section, we compare our results against random sampling and the multi-GS [211]. It directs the selection process of points for a sample towards inliers that exhibit the same residual, as the seed point to the models in a pilot set. In the guided sampling paradigm, the prevailing performance measure is the percentage of all-inlier samples. However, this is not a guarantee for a good putative model. The sample must consist of inliers that span the manifold. In some cases, a model generated from a sample of mixed points is closer to the ground truth model than from an all-inlier counterpart, provided that the signal to noise ratio in the sample set is high. Instead, we assign a putative model to the closest group of ground truth points. Then, we calculate their average residual. The value of the z parameter ranged from 50 to 500 in our experiments based on the sparsity of the test data, which was found to be adequate for the model deviation to stabilize. As shown in table 6.1, MST guided sampling consistently outperformed the rest. We point out that when applying MST guided sampling, there occurred some repetitions in the sampling sets. This redundancy favorably enhances the performance of the subsequent clustering algorithm.

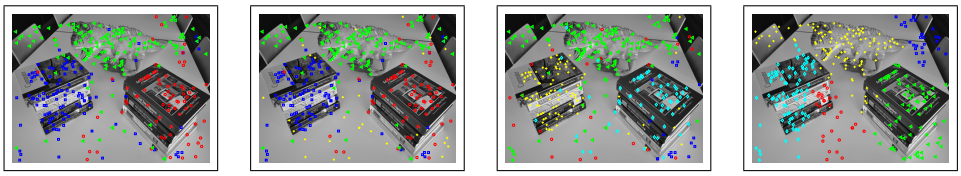
6.3.2 Multiplicity guided model detection

Figures 6.6, 6.7, and 6.8, show the results of our proposed algorithm, J-linkage [208] and PEARL [197] on the tested examples. We consider the classification power with respect to the data points of true models. For this purpose, we use the precision and recall values in table 7.2. In most cases of our approach, the precision values are lower than the other methods [208, 197], because, spectral clustering works by minimizing the cut in the similarity graph, thus maximizing the intra-cluster distance. In effect, this distributes the noise points in different clusters rather that aggregating them in a single outlier cluster. This increases the count of false positives per detected model. The precision figure is not alarming as long as the inclusion of outliers does not corrupt the models. This is proved by the enhancement in the recall figure, which shows its power in aggregating correct points to the models. This is due to the fact that our centroid finding technique is robust even in high levels of



(a) (b) (c) (d)

Figure 6.7: Merton college III. (a) Ground truth (outlier points added shown as yellow crosses). Results of (b) proposed algorithm; (c) J-linkage; (d) PEARL.



(a) (b) (c) (d)

Figure 6.8: Dino books. Result of (a) proposed algorithm; (b) post processing of outlier residual filtering increases precision to 0.7001; (c) J-linkage; (d) PEARL.

outliers, because it does not re-fit the model to the found cluster but rather elects its most probable model. We found that our algorithm always resulted in the correct minimal number of models. Such a goal has been advocated in PEARL [197] by the inclusion of a label penalty. In practice, however, the optimal choice of the label penalty is difficult.

6.4 Conclusion

We have presented a system for multi-model detection. We resolved to the guided sampling paradigm to hypothesize models and unprecedentedly used the MST, in this respect. Other novelties in our work include, a different perspective to defining goodness-of-fit, that depends on compactness of points and stability in the layout of models. Also, we proposed a model-to-model distance measure based on ranking that was shown to be very effective in describing model variation. The algorithm has the advantages of having no model-specific assumptions, being insensitive to model size variability and tolerating high levels of outliers. Finally, we showed that our algorithm outperforms the state-of-

the-art in multi-model fitting. The work presented in this chapter has been published in [22].

Chapter 7

Optimization of Facade Segmentation Based on Layout Priors

7.1 Introduction

Facade parsing is regarded as a classical case of semantic segmentation. As most scene interpretation approaches, the problem was originally tackled with appearance-based segmentation algorithms, in which weak priors of smoothness assumption are applied. Research was then directed to the incorporation of mid-level and high level cues of translational symmetry and sub-part classifications, based on training data. The challenges for achieving high accuracies arise from imaging artifacts: blur and noise, non-uniform lighting conditions, reflections and, shadows. Also, they include, the existence of irregular lattices of structures, occlusions, intra- and inter- geometric style variations, and the fact that some facade elements are looked upon as *stuff* [7] rather than *objects*. One way to deal with these challenges is to investigate the pattern of arrangement of facade elements in addition to their individual visual attributes.

In this chapter, we present an algorithm that exploits higher level reasoning about scene entities (superpixels), suggested by the appearance characteristics. We combine both aspects appearance and layout in a single energy function, to provide an optimized solution at the lowest level of image

primitives. The optimization function is solved via TRW-S algorithm [23]. It relies on a specific linkage between the superpixels based on translational symmetry and obtained by the α-expansion graphcut algorithm [121], in addition to a Region Adjacency Connectivity (RAG). In contrast, state-of-the-art methods [57] and [1] apply their optimization steps on formed Bounding Boxes (BB), whose assignments are either rejected or accepted as a whole. As such, their algorithms incorporates layout principles only in the recognition step of pre-segmented regions, resulting from appearance cues phase. Whereas, we carry out segmentation and recognition simultaneously, while exploiting the layout priors to correct preliminary segmentations. We provide an algorithm that minimizes the use of thresholds, prior assumptions except for fronto-parallelism and works in an approximate inference framework. More importantly, it does not require manual specification of architectural rules as in the 3-layered approach [57]. We achieve a reduction in the hard-coded parameters directly involved in the pixel classification due to the offline learning of scene statistics done on the training folds.

In chapter 3, we provided a comprehensive review of the work done in the field of facade. We complement it with a table of comparison (table 7.1) in which we place the work we are presenting in this chapter. This is to illutrate how our algorithm combines the most priors and in a pricipled way. We limit our table to research that reported accuracy scores on benchmark datasets. Also, we exclude the ones [113, 119, 118] that carry out the classification purely on appearance aspects of the structures with no inclusion of hints from layout. In addition, we eliminate grammar-based approaches [138, 132, 110]. Even though these works combine all layout aspects, layout integrity is enforced in a top-down direction leading to coarse subdivision. Grammars when compared to probabilistic approaches, are inflexible and for most part the grammar is manually specified or at least its learning is user-assisted. To recap, we include efforts that are fully automated, bottom-up, utilize a flexible form of architectural guidelines. An important factor in differentiating between proposed work that is application-specific, is the amount of semantic priors enriching the optimization function. Of course, there are aspects of algorithmic efficiency and computational cost. For this reason, we also specify the mode by which the layout guidelines are incoporated. Some priors are inter-dependent. For example, if translational symmetry is bound to x and y directions, then alignment is a frequent

	[1]	[57, 58]	[221]	[156]	[112]	Ours
Adjacency	●	○	♢	●	●	●
Spatial coherence	●	○	●	●	●	●
Location	♢	♢	♢	●		●
Translational symmetry	●	○				●
Alignment	●	○	♢	●	●	●
Vertical/Horizontal order	●	○		●	●	●
Non-straight boundaries					●	●

Table 7.1: Comparative table of the state-of-the-art bottom-up approaches in facade parsing. Legend: ○ indicates that layout prior is validated in a heuristic postprocessing step or via genetic-based algorithms, in contrast to ● which indicates principled non-heuristic involvement of the prior. ♢ states that the prior is quantified numerically and complements a feature vector that is fed into a classifier.

outcome. However, the converse is not true.

7.2 Facade Segmentation Optimization

Our proposed algorithm (Fig. 7.1) receives as input a set of image pixels in the $2d$ domain. It is required to provide an interpretation of these data points by assigning them to a predefined set of labels $\mathcal{L} = \{L_i\}_{m=1}^{M}$, such that $\mathcal{L}$ holds indices to M architectural structures. To keep the problem tractable and enhance computational efficiency, we work with superpixels. Thus, the data points for our algorithm is the set $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n}$ of n superpixels. The image is subjected to a watershed transform [28]. The transform aggregates pixels to a region until reaching a peak in the $2d$ space of the gradient image. The result is a severe over-segmentation of the images (figure 7.3) with color coherent regions, called *basins*. The superpixels are the minima pixels corresponding to the lowest gradient value in each region. Also, the use of superpixels enhances the extendability of the algorithm to cases of incomplete lattices, such as partially occluded images and $3d$ meshes. Because, the algorithm is no longer bound to grid linkage.

We pose our problem as an optimization problem under both appearance and layout constraints, emerging from architecture characteristic patterns. To this end, we define an energy function and minimize it using the sequential tree-reweighted message passing (TRW-S) [23]. Because our for-

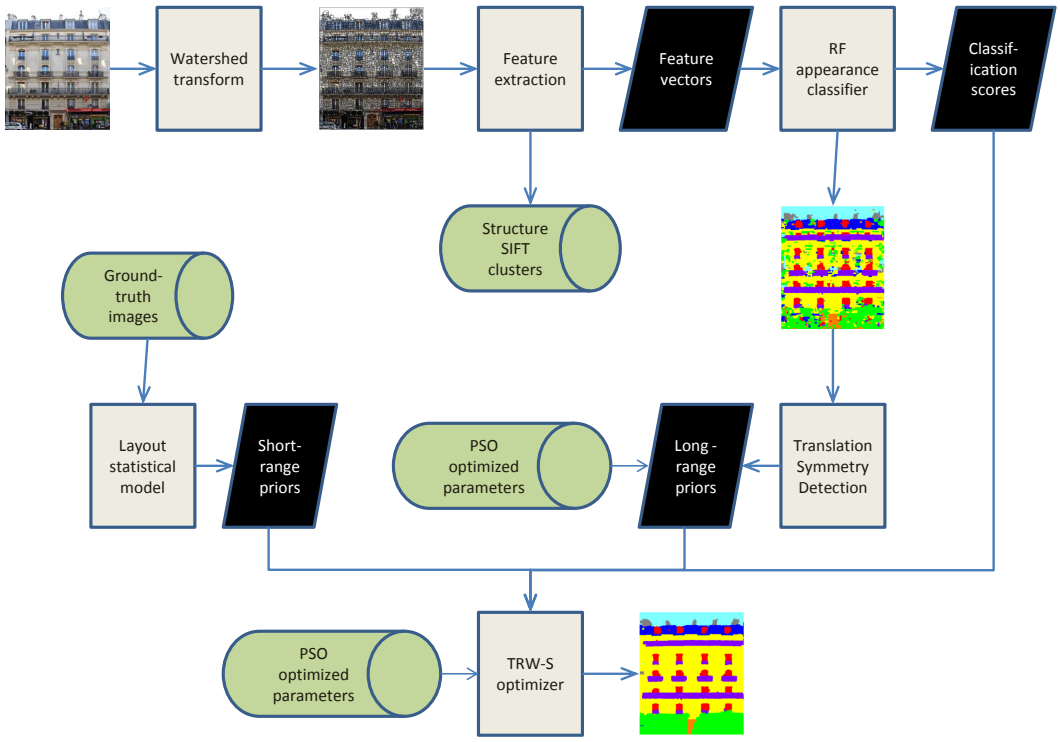


Figure 7.1: Diagram showing proposed system modules and their interactions.

mulation does not follow the sub-modularity constraints laid by α-expansion explained in chapter 4, we found that despite their widely acclaimed efficiency graphcuts will not accommodate the whole set of priors incorporated in our optimization function. We chose TRW-S due to its ability to handle arbitrary forms of cost function and scalability, while providing state-of-the-art results in some applications. We aim to ensure that the labeling of a pixel is influenced not only by the labeling of its neighbours, but also by that of pixels in other possibly distant regions based on extracted architectural patterns.

A distinctive aspect of our algorithm is imparting structural knowledge on image primitives. The TRW-S operates on the original set of superpixels. The total energy function Ξ of the TRW-S is as follows:

$$\Xi = \Xi_1 (L) + \Xi_2 (L) \quad , \tag{7.1}$$

where

$$\Xi_1 (L) = \sum_{x_i} D (L_i|x_i) \quad , \tag{7.2}$$

is the datacost received from the appearance module. $D (L_i|x_i) = -\log (P (L_i|x_i))$. $P (L_i|x_i)$, are the classification posteriors resulting from a Random Forest (RF) classifier. And, the layout prior

$$\Xi_2 (L) = \beta_1 \sum_{x_i} \sum_{x_j \in \Psi_1} Q_1 (l_i, l_j|x_i, x_j) + \beta_2 \sum_{x_i} \sum_{x_j \in \Psi_2} Q_2 (l_i, l_j|x_i, x_j) \tag{7.3}$$

is the total energy relayed from the layout statistical model and the translational symmetry modules (Fig. 7.1). Ψ_1 and Ψ_2 are the neighbourhoods defined based on the short- and long- range edges (Sect. 7.2.2), respectively. $Q_1 (\cdot)$ is the prior for the plausible structural adjacencies, while $Q_2 (\cdot)$ is the regularizer for the translational symmetry of structures in the architectural scene at hand. The assigned label of a superpixel is mapped to all pixels sharing its basin.

In the following sections, we explain how the appearance and layout priors are established to be incorporated in our energy function for the TRW-S algorithm.

7.2.1 Appearance Cues

A well-known fact about visual perception is, it is evoked by appearance. Thus, our algorithm is launched by obtaining preliminary classification of the image superpixels that utilizes textural characteristics of the regions. We choose Random Forest (RF) as our classifier [24], which performs a recursive partitioning of the data based on an ensemble of decision trees. But, other efficient classifiers can be used instead.

Another critical choice is the space in which the feature vectors are embedded. We examine 2 spaces. Firstly, the vector $\mathbf{s}_i$ is comprised of the 128 SIFT descriptors [38], calculated densely over the image with a bin size of 8. Secondly, the vector $\mathbf{r}_i$ (7.4) and (7.5) is the distances to M predefined clusters, corresponding to M architectural structures. Each cluster consists of the SIFT feature vectors of the superpixels, belonging to a certain structure and acquired from the groundtruth data. The distance is calculated as the mean Euclidean norm between the SIFT vector of the superpixel and the k-nearest neighbours vectors in the cluster after removing the exact match. We preferred this distance over a centroidal one, because the clusters exhibit a high degree of scattering, due to the high degree of appearance variation among instances of the same structure. Hence, the centroid would not be a proper representative of a cluster. we down-sampled over-sized clusters to ensure a uniform prior for the RF. In this way, the formation of the meta-feature vector involves both a non-parametric stage of calculation of distances to the groundtruth clusters and a parametric phase for inference through the learned Random Forest classifier.

$$\mathbf{r}_i = \left[r_i^1 \, r_i^2 \, \ldots\ldots \, r_i^M \right] \ . \tag{7.4}$$

$$r_i^j = \frac{\sum_{o=1}^{k} \left| \mathbf{s}_i - \mathbf{NN}_{ij}^o \right|}{k} \ . \tag{7.5}$$

$\mathbf{NN}_{ij}^o$ is the SIFT vector of the o-th nearest neighbour in cluster j with respect to data point i. And k is the count of neighbours.

In practice the later space was found to outperform the former. In our opinion, it introduced

a higher level of semantics over the raw SIFT features, that achieved a substantial dimensionality reduction (from 128 to M features). The challenge for any dimensionality reduction algorithm is, not disturbing the position of a feature vector in its space, relative to label clusters. In the described space, we retain this relative position of the vector, by storing its distances to the clusters in the space, without the overhead of low-level SIFT details. In addition, this space transformation provided better characteristics for the training vectors, namely inter-separability and intra-compactness of the clusters. These characteristics are expected to boost, not only k-nn equivalents but also margin-maximizing hyperplane classifiers. However, further investigation is required to evaluate the proposed idea with other classifiers and clusters of various topologies. Similar approach of using a meta-feature vector can be found in [9]. The resulting segmentations are provided as input to the next phase. We also retain the classification probabilities $P\left(L_i|x_i\right)$ computed by the RF for each super-pixel to be used as datacosts in the TRW-S framework.

7.2.2 Layout Cues

In this module, we make use of 5 architectural principles, namely, spatial coherence, approximate structural location, structure ordering, recurring structural adjacencies, and translational symmetry. In our framework, these principles are expressed in the edge costs of the TRW-S graph. The edge costs are look-up tables giving the penalties for various combinations of labellings for the edge vertices. There are 2 types of edges: short-range and long-range.

Short-range Edges.

They specify neighbours based on spatial proximity, and their edge costs used to establish $Q_1\left(\cdot\right)$ for the TRW-S function (7.3). Superpixels are connected by an edge if there is a common boundary between their encompassing basins. Hence, each superpixel is allowed a different number of neighbours. During the learning phase, we build a statistical model of the found adjacencies among structures. We argue that the familiar adjacencies is the most stable feature across different architectural scenes. For instance, A door structure can be seen adjacent to a wall, but never next to a sky

structure.

The edge costs are $M \times M$ matrices, where M is the number of architectural features. In POTTs model [123], the diagonal values are set to encourage neighbouring nodes get the same label. However, we utilize a non-POTTs model, in which the values on the diagonals of the cost matrices are non-zeros. We introduce the concept of *location-aware* edges, in which there is a penalty incurred even if nodes are given the same labeling. This penalty is dependent on the frequency by which the label has been seen in this zone of the image in the training samples. The frequencies of the labels with respect to locations are obtained through the following procedure. To account for image size variability, the groundtruth images are transformed to an approximate scale invariant space. This is done by subdividing the images into k horizontal and k vertical stripes of equal width, such that square patches are formed. The corresponding patch is determined for each labeled pixel and the information is used to update the frequency of the label in the patch. The values are then normalized by dividing by the total pixel density within the patch to encode the probability $\mathbf{P}_{rc}^{m}$, such that $r, c \in \{1, 2, \ldots k\}$.

To fill the upper and lower triangles of the cost matrices, we build a histogram (figure 7.2) similar to the one of the labels frequencies, but this time it is 2*d* for label transition on the same image subdivision. It encodes the structural tangencies. The recorded frequencies in each patch, are normalized per structure to reflect the probability $\mathbf{P}_{rc}^{ab}$ that a pair of labels (a and b) exist in adjacency at this location, when a testing sample is introduced. $a, b \in \{M \times M\}$, such that $a \neq b$. The edge costs are established in 2 directions corresponding to the directions for tangency: horizontal and vertical. For each structure instance in the ground truth, we record the structures to the east and south of it. We bypass the west and north directions because they are inverses of the included directions and would only require a transpose of the cost matrix. So, including them will redundantly duplicate the cost. We would like to point out, the matrices are non-symmetrical. For instance, a roof structure is more frequently seen to the south of sky than to its north.

In this way, the edge cost matrices (Fig. 7.3) encode the architectural principles of, vertical and horizontal arrangement ordering of structures, in addition to locations and structural direct adjacencies. At inference time, if basins are tangent in both directions, we choose the direction of the common

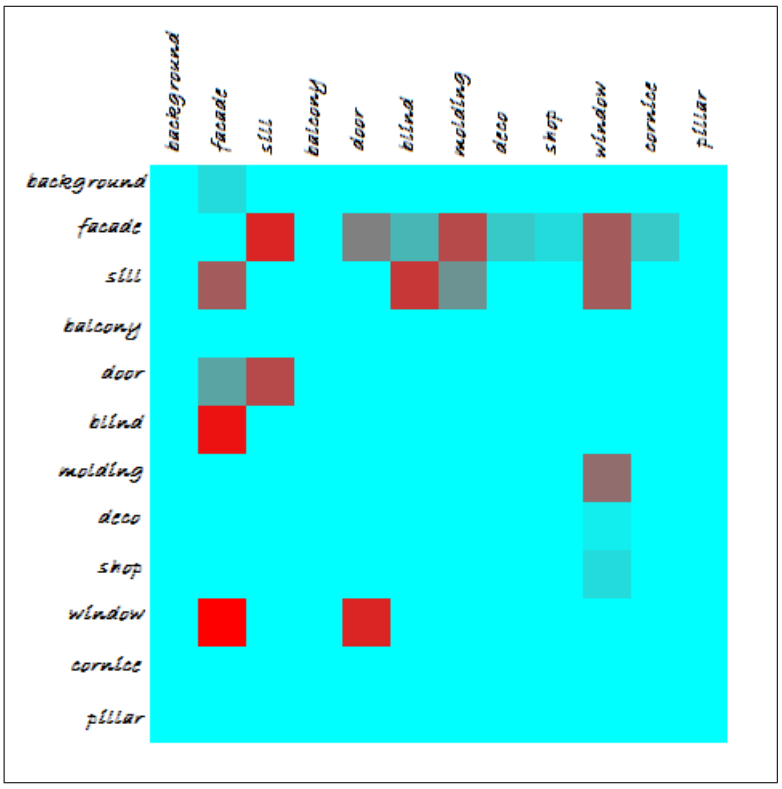
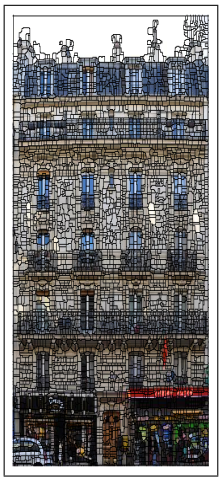
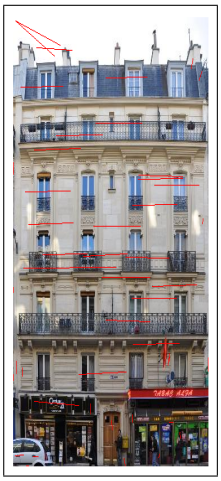


Figure 7.2: A sample of 2*d* adjacency histogram in the vertical direction. It indicates that the most adundant type of adjacency in this image patch is a facade underneath a window structure (indicated by the bright red color).



(a)



(b)

	1	2	3	4	5	6	7	8	9	10	11	12
1	1	21	464	464	21	464	464	464	464	464	21	21
2	464	2	21	464	21	464	21	464	21	21	21	21
3	464	464	7	464	464	464	464	464	464	464	464	464
4	464	464	464	444	464	464	464	464	464	464	464	464
5	464	464	464	464	5	464	464	464	464	464	464	464
6	464	464	464	464	464	444	464	464	464	464	464	464
7	464	20	464	464	464	464	5	464	464	464	464	464
8	464	464	464	464	464	464	464	444	464	464	464	464
9	464	21	464	464	464	464	464	464	5	21	464	464
10	464	21	464	464	21	464	464	464	464	4	464	464
11	464	464	464	464	464	464	464	464	464	464	444	464
12	464	464	464	464	464	464	464	464	464	464	464	6

(c)

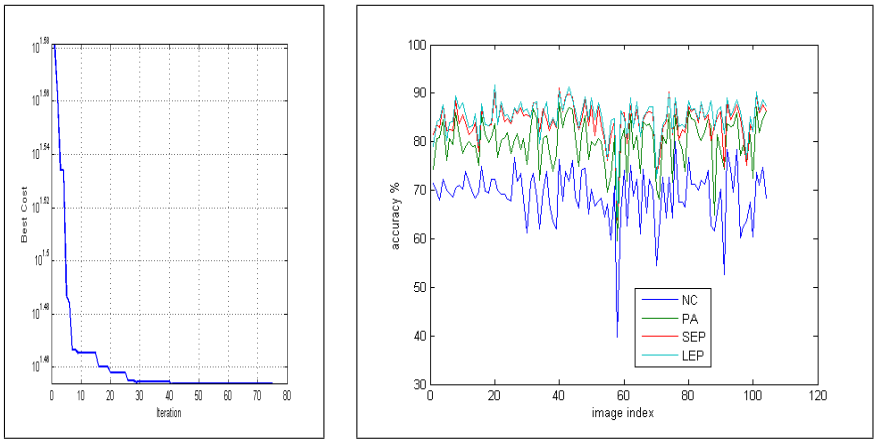
Figure 7.3: (a) A sample of a watershed segmented image that reveals how small the superpixels are. (b) A sample of long-range edges shown in red. (c) A sample of short-range edge approximated cost matrix for the CMP dataset [1]. Structure 1 incurs the least cost, which signals that it is the most frequently encountered structure in this image patch. The most abundant transition is between structures 7 and 2. Structures 4, 6, 8, and 11 are never seen in this image patch during training. Values on the diagonal are in a lower range than the ones on the lower and upper triangles to promote same labeling.

boundary with the longest length. We convert the probabilities to costs to build labeling penalty matrices, according to the Boltzmann distribution, $\mathbf{E}_{rc}^{m} = -\log\left(\mathbf{P}_{rc}^{m}\right)$ and $\mathbf{E}_{rc}^{ab} = -\log\left(\mathbf{P}_{rc}^{ab}\right) + \xi$. We add ξ, a constant to raise the range of values in the upper and lower triangles of the cost matrices over the diagonal values, to bias the optimization algorithm towards same labeling for the vertices of the edge. As such, spatial coherence is achieved while promoting the frequently encountered label in the training set, at this location. If the algorithm chooses to label the vertices differently, the most frequent adjacencies at this location are preferred.

Some practical adjustments were carried out, because the subdivision of the image is arbitrary and to prevent over-fitting to training data. We apply a Gaussian smoothing filter on the frequency histograms of location and structural adjacency. In addition, Inf costs, resulting from zero frequency, are replaced by a relatively high value π, to discourage rather than eliminate the possibility of an assignment. Same goes for Inf values in the appearance datacost, as they are replaced by ρ.

Long-range Edges.

These encode the translational symmetries found in the scene, used for building the $Q_2(\cdot)$ (equation 7.3). To establish these symmetries we use the α-expansion graphcut algorithm [121], to assign a translation vector to each superpixel in the image. The ultimate goal is to establish a smoothness prior over distant instances of the same structure, in the TRW-S labeling optimization step. It is run separately for each type of putative structure resulting from the appearance classification phase. A Markov Random Field (MRF) is defined over all superpixels belonging to the structure and forming the nodes of the graph. The smoothness prior is based on neighbourhood Ω, detected between superpixels when their basins share a common boundary and belong to the same putative structure. Neighbourhoods are assigned a constant weight. The terminal nodes of the graph of the α-expansion algorithm constitute the labels and they are a set of translational vectors. This set is constructed from the SIFT feature points of the image and their best matches within the putative structure. The matching score is calculated based on Euclidean norm in the SIFT space. The set of translational vectors is refined by preserving only the ones that exhibit a translation in either the *x* or *y* directions but not both, as we are assuming pre-rectified images. As such the long-range cliques promote the vertical and horizontal alignment of facade structures. Also, we eliminate the possibility of assignment to a translation that maps a basin to itself. In the set of translations, there are ones that are exact replicas of each other but in the opposite direction. This happens when the start and end points of the translation are exclusive best matches of one another. If left without handling, this unnecessarily assigns units on opposite parts of the same lattice to various labels differing only in direction, and causes a conflict for the parts in the middle, under the smoothness prior. After removing such replicas, we calculate the datacost based on the resulting set of translation vectors in 2 directions, the originally found one and its reverse (a 180° rotated variant). For each point-to-label cost, it is determined as the magnitude of the minimum value of both directions.



(a) (b)

Figure 7.4: (a) Semi-log scale plot of the cost against PSO iterations. (b) Accuracy plots for the images in ECP dataset when different options for IASC are activated.

The energy function E, to be minimized by the graphcut, is as follows:

$$E\left(Y\right)=\sum_{x_i}D_Y\left(y_i|x_i\right)+\mu\sum_{x_i}\sum_{x_j\in\Omega}F_Y\left(y_i,y_j|x_i,x_j\right)+\theta\cdot|Y_T|\ . \tag{7.6}$$

The unary term $D\left(\cdot\right)$ is the dissimilarity score between an examined superpixel x_i and the superpixel of the watershed basin, to which the destination belongs $\overline{x}_i$. The destination is obtained when applying translation y_i ($\in T$) on the examined superpixel. We constraint the translations to result in destinations being within image boundaries. The dissimilarity is defined by the Euclidean norm $\|\cdot\|$ and the $f\left(\cdot\right)$ corresponds to the SIFT features at the point (equation 7.7). The pairwise term $F\left(\cdot\right)$ follows a POTTS model, in which a pair of neighbouring superpixels labeled differently, is penalized with a constant value. θ is a constant label cost that penalizes the assignment of x_i to new redundant labels. Redundancy in the sense that they can be replaced by one of the already utilized labels without drastically increasing the datacost

$$D\left(l_i|x_i\right)=\|f\left(x_i\right)-f\left(\overline{x}_i\right)\| \tag{7.7}$$

After the optimization is carried out and the superpixels are assigned to translational labels, the actual linking of superpixels is carried out by implementing both the translation and its rotated variant on the source superpixel. This is done regardless of whether the destination will be inside the putative structure or not. In effect, this extends the putative structures into a loci of points that complete their contained grids. This was found to be helpful in reducing the propagation of limiting errors from the previous appearance-based stage in deciding putative structures. An outcome of this phase is shown in Fig. 7.3. The formed edges will be relayed to the TRW-S algorithm.

To sum up, superpixels are now linked by an edge, if they were adequately similar in SIFT space and their spatial neighbours vote for this link. More importantly, if they were part of a hypothetical extension of the found grids. A limitation encountered in the literature is that each genre of architectural element is allowed to contain only one grid. In our formulation, each putative structure may contain multiple grids as superpixels are free to belong to any translational label.

Establishing neighbours only within putative segmentations enabled us to increase θ and μ to a level that encouraged whole Connected Component (CC) translation and electing the most abundant vectors in the structure. This is not easily attainable with a color weighted neighbourhood in a 4/8-connectedness setting without the risk of structure boundary over-smoothing. If an edge occurs twice based on both short- and long- range cliques, spatial prior takes precedence and the long-range connections are dropped.

7.2.3 Learning the Hyperparameters.

Cross-validation is the dominant approach towards assigning values to the hyperparameters of the energy function. It is an indirect technique to the calculation that involves subdividing the dataset into folds and experimenting with different user-assigned values for the different folds. This is a matter of probing the parameter space to detect promising regions. In some work, there is a further refinement step by employing the local search technique of gradient descent for optimizing values. However, the assumption of strictly compact convexity of the parameter space associated with gradient descent, limits its application to wider ranges. In the cross-validation, to avoid getting trapped in local minima,

the probing action should be extensive. In effect, this performs an exhaustive search of the parameter space. The approach is suitable when the space is limited in terms of its degrees of freedom, bounding range of each parameter and the precision of the values (i.e. whether they are discrete or continuous). It scales poorly with any added complexity. Generally speaking in the reported experiments utilizing this technique, the count of hyperparameters does not exceed 3. In our case, it reaches 5. We resort to evolutionary algorithms. Truly, they are time consuming algorithms, but in our case the procedure is conducted once offline prior to the commencement of the processing of our optimization function. We use the Particle Swarm Optimization [25] (PSO). A meta-heuristic technique, that relies on a user-specified range of values for finding the parameters. The main merit of this approach is that it exploits knowledge captured through previous calculations of the cost function to make an informed decision about the direction of search for the optimal values. This is in contrast to cross-validation which perform independent runs for the values. At the same time, PSO tries to avoid local minima by incorporating a randomness element in its update equation. It also holds another property that enhances its robustness to minima. Even unpromising particles (possible set of values of the hyperparameters) are allowed to continue processing spreading more extensively over the solution manifold. They iteratively enhance their positions while being able to globally communicate their best positions so far to other particles. This leads to propagation of knowledge about the different subspaces of the parameters manifold and enhances the convergence of the algorithm.

PSO initializes a swarm of vectors randomly. Each vector U_i holds values for the parameters and is named a particle. Iteratively, it updates the vectors based on their best previous position $U_{\mathrm{i\_pbest}}$ and the best position in the swarm $U_{\mathrm{global\_best}}$. The quality of the particle is evaluated based on a cost function. In all our experiments, the cost function is single objective. However, PSO has been extended to handle the multi-objective case [222] and has been used in significant applications [223]. The position update rule for the i^{th} particle is

$$U_i = U_i + V_i \ . \tag{7.8}$$

The velocity V_i of the particle is given by,

$$V_i = \omega \times V_i + c_1 \times \text{rand}\,() \times (U_{\text{i\_pbest}} - U_i) + c_2 \times \text{rand}\,() \times (U_{\text{global\_best}} - U_i) \quad . \tag{7.9}$$

The rule guarantees that the procedure yields non-increasing cost values in each iteration Fig. 7.4, thus leading to convergence. First, we use the PSO in learning the α-expansion parameters (θ and μ). In this case, the objective is minimizing the number of erroneous edges that link superpixels belonging to different genre of structures. In the second setting, it is used for optimizing β_1, β_2, ξ , π, and ρ in the TRW-S framework. Intuitively, the ultimate criterion for the goodness of a segmentation proposal based on the current set of weights, is its coherence to what is chosen to be the best perceived by humans. Thus, we use a supervised scheme for evaluating the goodness of the proposed segmentation based on ground truth.The objective is minimizing the errors in the final labeling of the superpixels, when compared to ground truth data.

7.3 Evaluation

We follow the convention of related work, and document the results based on 5-fold cross validation and using pixel-based accuracy as the criterion for comparison. The training folds are used for constructing SIFT clusters of the structures, collecting the layout statistics and training the Random Forest. We test our model IASC (Integrated Appearance Structure Cues) on the *ECP-Monge* dataset [224] and the *CMP* dataset [1], and compare to the *state-of-the-art* results from the 3-layered approach [57], Spatial Pattern Templates (SPT) [1] and Auto-Context [120].

The utilized datasets are benchmarks of the field. Both are fully annotated allowing for supervised training mode. Facade cropping and rectification are predominant assumptions in the field. For this reason, the datasets are preprocessed as such.

The *ECP-Monge* contains 104 images of facades in Hausmannian style. There are 8 structures specified in the groundtruth maps. We use the corrected ground truth [57]. The set of possible labels $\boldsymbol{\mathcal{L}}$ is $\{window, door, shop, wall, roof, balcony, sky, chimney\}$.

	SPT [1]			3-layers [57]			Auto-Context [120]		IASC (our method)			
	(NC)	(AP)	(APRT)	(NC)	(PA)	(SH)	(ST3)	(PW3)	(NC)	(PA)	(SEP)	(LEP)
ECP	59.6	79.0	84.2	82.6	85.1	84.2	90.8	91.4	68.9	79.9	86.3	87.8
CMP	33.2	54.3	60.3	-	-	-	66.2	68.1	41.4	55.5	60.3	64.4

Table 7.2: Average accuracies on datasets. NC: No context (appearance only), AP: Aligned Pairs, APRT: Aligned Pairs Regular Triplets, SH: Structural Heuristics, PA: POTTS Adjacency, ST3: Auto-Context classified, PW3: POTTS Smoothed Auto-Context, SEP: Short-range Edges Prior, and LEP: Layout Edges Prior (short- and long- range).

The *CMP* dataset is considered more challenging as it contains 378 samples with 12 structures from various (often difficult to model) styles. The set of possible labels $\boldsymbol{\mathcal{L}}$ is

$\{background, facade, sill, balcony, door, blind, molding, deco, shop, window, cornice, pillar\}$.

Because, we propose a multi-phase algorithm, we needed to separately examine each phase to understand its contribution to the final accuracy value. Table 7.2 summarizes the mean accuracies achieved by [57], [120] and [1] and IASC algorithm in various stages. We include the results of the commonly used POTTS model for spatial smoothness (PA), as a variant of our algorithm, and use the same datacosts of the IASC. We follow the naming conventions of the original papers [1, 57, 120] in reporting results. Per-image accuracies are shown in Fig. 7.4, for the different factors affecting the performance of our model. In Fig. 7.5, we display results of a selection of samples. It is comprehendible that the dataset with a unified architectural style achieves the higher accuracy. The monotonicity of style involves stability of the textures, in addition to reasonable stability with respect to building size, number of floors and arrangement of elements, which boosts the effect of the learned layout statistics. We can conclude from experiments and the reported accuracies in 7.4, for IASC each phase consistently improved performance over the preceding one. That is to say, each phase that involved the addition of a new structural prior and, invariably increased accuracy for all samples. Despite our efforts to minimize the propagation of errors, across the system modules, it is evident that appearance classification failures remain a limiting factor for subsequent improvements. Images that were badly classified in the appearance phase remained as the worst outputs even when structural knowledge was used. It is evident for [57], the incorporation of the structural heuristics

(such as: the existence of a running balcony on the second and fifth floor) degraded the accuracy of their smoothed appearance classifications. As for [1], the fact that their neighbourhoods of pairs and triplets were based on a manually assigned threshold was a severe limitation. The reported result for ECP-Monge in [120] is based on 7 classes of structures, whereas we include the result using the updated groundtruth which added the chimney structure. In IASC, we record one of the highest accuracy net gains when incorporating layout cues in the problem of facade parsing, even when starting with severely damaged results based on appearance. This is attributed to the generalization ability of our optimization function that relies only on persistent architectural guidelines without being style specific. Results are shown in figure 7.5.

We use the Davies Bouldin (DB) index [135] to shed light on the characteristics of the proposed feature space of distance-to-cluster, against the raw SIFT feature space. The clustering is predefined from the groundtruth and we normalized the 2 spaces. It was found that the proposed space transformation increased both separability and compactness of the clusters, thus, favorably lowering the average DB on the training folds from 8.4616 to 1.4497. As for classification accuracy, raw SIFT vectors achieved 63.3% on ECP-Monge in the No Context setting. For the distance vectors, the figure was 68.9%.

In both settings we use the PSO to learn the parameters, the number of iterations was set to 75. The swarm size was 10 when optimizing the parameters for finding long-range edges and 40 for the TRW-S function. The parameters ranges were based upon our observations during experiments, but we provided a much wider range to lower the risk of a local minimum. In evaluating the objective functions, 10 samples were selected randomly for each dataset. The objective function is determined as the maximum calculated cost resulting from the 10 samples, which placed an upper-bound on the number of errors.

We tried Delaunay triangulation, to establish neighbourhoods between putative superpixels of a structure, in the translational symmetry module. However, the induced complete linkage degraded the assignment, by relying on the votes of *distant* superpixels - correctly or erroneously classified in the appearance phase.

We would like to point out that algorithmically, the approach is fully fit to be implemented at the pixel level. We have presented it while utilizing superpixels only for computational cost. Computers of higher power and memory should be able to handle this. However, we did not find significant accuracy gains in the pilot experiment we carried out due to the fact that the superpixels are very small and approximately of uniform color. Very few have handled facade analysis with layout priors at the pixel level. To our knowledge, only[120, 112] have done so. However, in [120] the layout cues are found in the form of numeric values in feature vectors which are sequentially and independently classified. While, our algorithm provides concurrent labeling of the image primitives leading to a more globally optimal result.

7.4 Conclusion

We present an algorithm for handling semantic segmentation of architectural scenes. The algorithm relies on the output of a Random Forest classifier on SIFT-based meta-feature vectors. We carry out a feature space transformation from raw SIFT to distance-to-cluster vectors. Also, we incorporate layout principles in the form of labeling costs for superpixel long-range cliques resulting from translation vectors, detected by α-expansion. Other labeling costs are based on location and structural adjacencies, defined on short range neighbourhoods. We report competitive results. We believe our method offers significant advantages over competitors in terms of algorithm elegance. The priors are automatically learned from training samples and its weight parameters are deduced via the single objective PSO algorithm. At inference time, the labeling is efficiently optimized using the TRW-S algorithm, while including no heuristics or manually determined thresholds. Also, we impose the priors on superpixels resulting from severe over-segmentation, in contrast to, the common practice of optimizing the labeling of whole structures with no ability to fine tune at a pixel/superpixel level. The work can be found in [26].

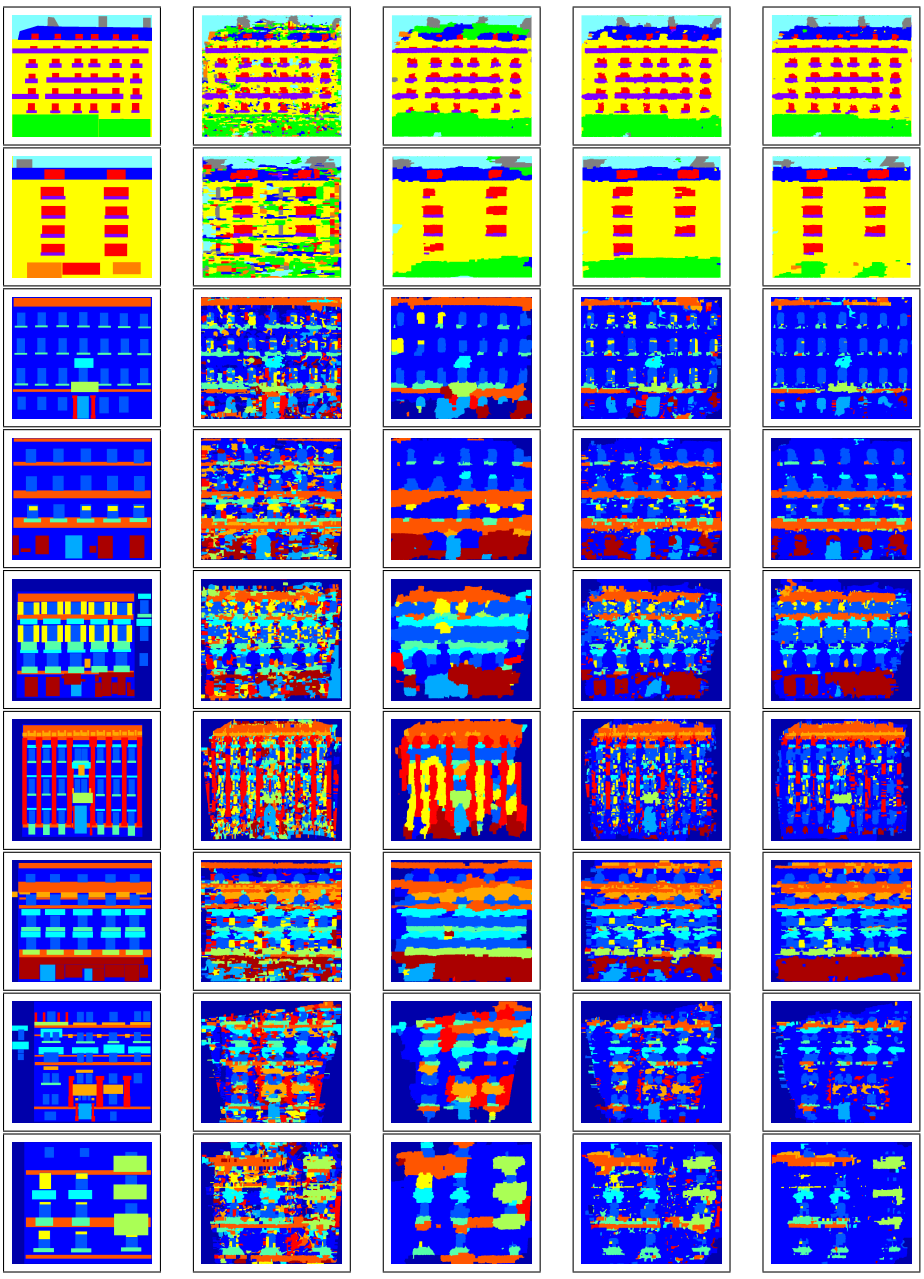


Figure 7.5: Sample outcomes in tabular format. Rows from (1) to (2) ECP-Monge samples; Rows from (3) to (9) CMP samples. Column (a) Ground truth; results of (b)NC; (c) PA; (d) SEP; (e) LEP.

Chapter 8

A Deep Learning Pipeline for Semantic Facade Segmentation

8.1 Introduction

Most previous work in facade parsing is based on embedding appearance and contextual constraints in an energy function solved by efficient optimization techniques, such as factor graphs and integer programming. The appearance prior is usually handled by applying pre-trained classifiers on SIFT variants. It forms the unary potential in the energy formulation. Contextual priors are then enforced through the higher order potentials. They encode the structural guidelines of the facades such as, encouraging the existence of equidistant alike structures and penalizing impossible adjacency relationships. For example, a roof can never be adjacent to a shop structure. It is evident, there is a need to move onwards to techniques that alleviate the need for hand-engineered features and architecture specifics.

In our work, we develop a pipeline which relies on 2 deep learning techniques, namely; Convolutional Neural Network (CNN) [225] and RBM [16]. Deep learning techniques provide the most automated and highest performing approaches to computer vision problems. They are scalable and work directly on raw visual data,without the need for manual specification of (hyper) parameters nor

classification guidelines. CNNs are acclaimed for achieving state-of-the-art results on the PASCAL VOC 2012 benchmark [104]. We utilize them in the classification module based on appearance qualities of the regions. The output is refined by another fine-grained classification module that uses RBMs to enforce contextual constraints.

Using context at pixel level classification to solve appearance ambiguities, has been a subject of research. The difficulty has been twofold. Firstly, representing the layout in a feature space that is more computationally efficient than the raw pixel space which normally requires a dimensionality reduction, while being able to preserve characteristics of the scene . This is to ensure that the alternate representation has captured the essence of the image. Secondly, the ability to assess the share of each pixel in the global layout, in a way that allows a local decision in fine grained vision tasks. Several efforts can be found in [226, 227].

Our use of the RBM is an attempt to tackle the aforementioned difficulties. We utilize its generative ability to restore the true structure of the scene. This ability has been used in solving the occluded parts problem [228]. In our formulation, we are relying on votes from the accompanying architectural structures in the arrangement to recover the erroneous classifications resulting from the CNN.

We propose an algorithm that is conceptually novel in various aspects. The algorithm maintains a global outlook to the scene while being able to fine-tune the final classifications at the pixel level. This is in contrast to the norm in the literature, which only refines classifications of preliminary whole structures. In addition, it builds its labeling on 2 models; the one based on experience from past data and a model of the captured layout of the scene at hand. This allows flexibility and extends the generalization ability of the trained machines. Moreover, we are refining the structural decision making tool that will be used to update the preliminary results to yield the final results.

Perhaps, the most related to ours is the work in [229]. They also produce probabilistic image segmentation initialized by CNN softmax posteriors. They utilize Conditional RBMs (CRBMS), which are the discriminative extension of basic RBMs used for structured output prediction. Their architecture involves addition of another set of nodes corresponding to the observed set of features

and which constitute the input to the machine. The output is obtained after the processing on the visible nodes. In contrast, in our formulation the input and output is fed and obtained respectively via the visible nodes. The change in architecture incurs a completely different learning algorithm. CRBMs are trained through mean-field inference, a message passing algorithm. While, in our work we follow a CD-based training.They report achieving an increase of 0.6% only when jointly training the CNN and RBM, in the simplified application of segmentation of cropped images of faces and animals. This percentage increase is not motivating for us to apply their technique in our application, where a more complex type of scenes is expected. In the literature, it is obvious that so far applying shape RBMs have been confined to cropped images where there is a single focus-of-attention object. In our algorithm, we provide an application of them in a more complex genre of scenes.

8.2 Proposed Algorithm

The input to our algorithm is a RGB-valued facade image I. The aim is to provide an interpretation of the scene into meaningful architectural structures. Formally, the algorithm receives as input a set of image pixels $\mathbf{D} = \{\mathbf{d}_n\}_{n=1}^{N}$ in the $2d$ space. $N = r \times c$, where r and c are the vertical and horizontal dimensions of the image respectively. The algorithm classifies these data points by assigning them to a predefined set of labels $\boldsymbol{\mathcal{L}} = \{L_m\}_{m=1}^{M}$, such that $\boldsymbol{\mathcal{L}}$ holds indices to M architectural structures. We present a deep learning pipeline that utilizes both appearance and structural aspects of the scene. The core of our algorithm is the use of RBMs, to enforce architectural constraints deduced from past data and then to learn the structure of the test image at hand to allow it to make pixel labeling decisions based on the majority of its own pixels. The RBMs cascade is initially stimulated by predictions collected from a deep convolution network. Please refer to figure 8.1 for an overview of the algorithm.

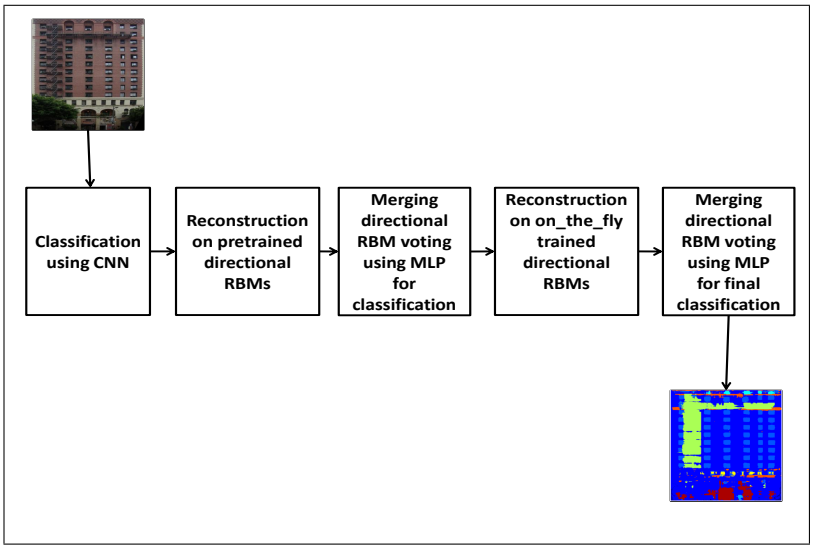


Figure 8.1: A schematic showing system modules

8.2.1 Appearance cues

We have utilized the VGG-model [89] of a CNN, adapted to the task of pixel-wise classification [95] and pretrained on ImageNet [6] dataset. The adaptation involves replacing the fully-coonected layer at the top of the VGG-model with a $1 \times 1 \times M$ filters to reflect class posteriors. Transfer of knowledge is a concept introduced in [230]. It is the core of using pretrained networks. It implies that filter parameters learned when training on one dataset can be used to launch training on another and is especially beneficial when the second dataset is small-sized. The main features of the architecture is relying on small filter sizes that range from 1×1 to 3×3, to restrict the number of parameters to learn and prevent overfitting. Also, there are deconvolution and interpolation layers to restore the original image size. In addition, lower level features and features from higher levels are combined through specialized *skip* layers and propagated ahead in the architecture. Learning the network parameters is based on minimizing the cross-entropy between the network outcome and ground truth.

The outcome of this phase is I^0 an appearance-based multinomial labeling map for the image, obtained through a softmax operation on the CNN classification scores.

8.2.2 Structure cues

The elimination of the fully-connected layer of the VGG-model degrades the CNN ability in capturing a global layout of the scene. It is true, pooling and window filters cause the CNN in the topmost layer to gain a wider scope of the neighborhood of each pixel. But, it does not extend to include the whole/near-whole scene while involving fine-grained classifications in any currently available architecture. In addition, the weight sharing concept makes the CNNs neutral to the location of the assigned labels due to the achieved translation invariance. More importantly, CNNs perform best when presented with signals with a rich range of frequencies. This is the reason it is suitable for dealing with image textures. However, when the study involves the inter-relations between architectural elements expressed as pixels labels, such that equidistance and repetition are discovered another technique would be required. Inspired by the work in [231], we opt for the generative probabilistic model of RBMs for learning and enforcing architectural guidelines so that the preliminary labels are refined. A RBM is mainly utilized to learn the joint probability distribution of the data at its visible nodes. Thus, by clamping the pixel labellings to the visible nodes, the machine learns the associations between the labels and consequently the inter-relations between the architectural elements. We extend the RBM-based model from recognizing whole images to fine-grained recognition of image regions in a similar way to the Shape Boltzmann Machine (SBM) [175] and its extended version to multi-part objects (MSBM) [178]. However, our utilized visible nodes are binary instead of the SBM multinomial nodes. This is to preserve the original learning rule of the RBM formulation and retain its convergence properties [232]. In the following, we present a generic formulation of the utilized RBMs throughout this work. The RBM consists of 2 sets of nodes; namely the visible $\mathbf{v}$ and hidden nodes $\mathbf{h}$. The interconnections are symmetrical and the intra-connections are not allowed. The restriction imparts on the graphical model properties of tractability with respect to the calculated distributions. $|\mathbf{v}| = |\mathbf{h}|$. According to [16], adding hidden layers with the same number of nodes as the visible layer is guaranteed to lower error rates.

The model finds a joint probability distribution for $\mathbf{v}$ and $\mathbf{h}$ that can be represented as a Gibbs

distribution of the form,

$$P(\mathbf{v}, \mathbf{h}; \theta) = \frac{1}{Z} \exp\left(-E(\mathbf{v}, \mathbf{h}; \theta)\right) \tag{8.1}$$

where, Z is the partition function

$$Z = \sum_{\mathbf{v}} \sum_{\mathbf{h}} \exp\left(-E(\mathbf{v}, \mathbf{h}; \theta)\right) \tag{8.2}$$

and E is the joint energy

$$E(\mathbf{v}, \mathbf{h}; \theta) = -\mathbf{h}^T W \mathbf{v} - \mathbf{a}^T \mathbf{v} - \mathbf{b}^T \mathbf{h} \tag{8.3}$$

$\mathbf{W}$ and $\{\mathbf{a}, \mathbf{b}\}$ are the weight tensor and the set of visible and hidden biases, respectively. Collectively, they comprise the set of network parameters θ. During training, θ is optimized to maximize the log-likelihood of the visible data, by minimizing the discrepancy between this data and its reconstructed version. We use the FEPCD sampling technique introduced in [233] to approximate the derivative of the log probability of the training data, in the stochastic gradient descent process used to optimize θ. Similar to PCD [184], multiple persistent Markov chains are maintained to approximate the model-dependent expectations of the data. However, contrary to the PCD, there is a deterministic selection of the chains based on the free energy of the visible vector. The authors argue that the samples with the lower energy adhere better to the model's distribution, as they compute the likelihood gradient more accurately. Thus, the algorithm retains only half of the chains having the lowest energies.

In all settings, $v_i \in [0, 1]$ and $h_j \in \{0, 1\}$ and the conditional distributions defined over them are given by

$$p(v_i = 1|\mathbf{h}) = \sigma\left(a_i + \sum_i w_{ij} h_j\right) \tag{8.4}$$

$$p(h_j = 1|\mathbf{v}) = \sigma\left(b_j + \sum_i w_{ij} v_i\right) \tag{8.5}$$

where, $\sigma(\cdot)$ is the logistic function.

We use the probability $p_i = p(v_i = 1|\mathbf{h})$ instead of sampling binary values. According to [232], this reduces the sampling noise and boosts the learning speed. In addition, the value p_i will be

regarded as the likelihood of having a certain label L_m at some pixel.

Algorithm 1 Structural Inference

Require: $I^0, R^y_\Phi, R^x_\Phi, B$
  $R^y_\Psi initialized, R^x_\Psi initialized$
  for each $k \in \{y, x\}$ **do**
    $I^0_k \leftarrow resize(I^0, s_k)$
    $\Gamma_{I^0_k} \leftarrow binarizeScanlines(I^0_k)$
    $\Gamma^\Phi_{I^0_k} \leftarrow reconstruct(R^k_\Phi, \Gamma_{I^0_k})$
  end for
  $T \leftarrow formMetaFeatures(\Gamma^\Phi_{I^0_y}, \Gamma^\Phi_{I^0_x})$
  $I^1 \leftarrow predict(B, T)$
  for each $k \in \{y, x\}$ **do**
    $I^1_k \leftarrow resize(I^1, s_k)$
    $\Omega_{I^1_k} \leftarrow binarizeScanlines(I^1_k)$
    $\hat{\Omega}_{I^1_k} \leftarrow augment(\Omega_{I^1_k})$
    $R^k_\Psi \leftarrow train(R^k_\Psi, \hat{\Omega}_{I^1_k})$
    $\Gamma^\Psi_{I^0_k} \leftarrow reconstruct(R^k_\Psi, \Gamma_{I^0_k})$
  end for
  $T \leftarrow formMetaFeatures(\Gamma^\Psi_{I^0_y}, \Gamma^\Psi_{I^0_x})$
  $I^2 \leftarrow predict(B, T)$

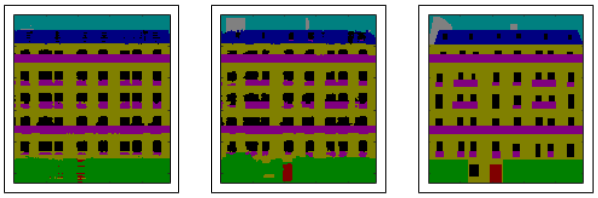
Layout validation

We have trained 2 specialized RBMs: R^y for vertical and R^x horizontal scanlines. For each direction, the ground truth labeled image $G \in \Phi$ is resized to a fixed dimension (s_y for the vertical and s_x for the horizontal), while leaving the other dimension as a free parameter in order to preserve the aspect ratio. We use nearest neighbour interpolation for the resizing. The result is two transformed images G_y and G_x. G_y produces $floor\,(s_y \cdot (c/r))$ scanlines of length s_y and G_y produces $floor\,(s_x \cdot (r/c))$ scanlines of length s_x, where r and c are the original height and width of the image. Basically, a scanline is a vector of pixel labels. The directional scanlines are accumulated into 2 training sets for the R^y_Φ and R^x_Φ RBMs.

Each scanline is binarized. This means, the q^{th} pixel on the scanline is represented by a one-hot mini-vector $o^q_\mathcal{L}$ with the value 1 at its label index. This is similar to the approach found in [229]. The

mini-vectors of all pixels on a single scanline are aggregated in one flat vector. The visible data Ω_{G_y} and Ω_{G_x} for the RBMs are the collections of these flat vectors in each direction. As such the machine, when trained on these scanlines, learns the associations between different labels at different aligned locations concurrently along columns and rows.

The approach of image subdivision can be found in [175]. It is normally carried out to keep the computation burden within tolerable limits and to escape severe resizing that might destroy the image layout. More importantly, it allows the RBM to focus on the highly stable pixel interactions which are mostly local. In our application, we opt for the scanline subdivisions as they hold the essence of architectural scene global semantics. They encode cues of structure order, neighbouring relations, equidistant repetitions, approximate location and alignment. In addition, as the training assumes independence between scanlines, we are implementing the weight sharing concept, commonly seen in CNNs, which achieves translation invariance. Thus, the location is no longer a coordinate value (even in normalized form) but rather a gestalt voting that emerges from the majority of pixels labels on the scanline. This tackles the scale-space difficulties encountered in location dependent approaches and boosts the algorithmic ability to deal with cropped images of facades. We regard our formulation based on scanlines as the first to tackle the problem of imposing layout constraints on scenes using RBM. In [178] the quadrant subdivision is confined to cropped images of single objects in focus (horse, face). The added complexity of scenes has multiple implications handled by scanlines. Severe resizing of the image in order to get the image quadrants to a size comparable to our scanlines, would devastate the layout of the building and render some structures unseen even by human eye. Generally, in context related applications, the challenge is using the largest scope of the image while maintaining the problem in reasonable size. Our formulation allows the decision at a single pixel to use knowledge from the whole span of the image width and height, thus achieving a higher degree of globalness and within tolerable computational burden. It encodes arrangement relations between more structures and allows repetition patterns to be more evident. Whereas, a quadrant subdivision will probably miss the symmetry of windows in a grid and miss encoding relations between structures inherent to different zones (sky and shop). In addition, our subdivision is the most favorable approach to handling the



(a) (b) (c)

Figure 8.2: Final labeling of a sample facade image (a) without dataset augmentation, (b) with dataset augmentation. (c) The groundtruth map. It is clear that the door and chimney structures were correctly recovered in (b)

small-sized dataset. A single image contributes with numerous scanlines raising the count of training instances from a few hundreds to thousands. This is done while preserving a wide scope of the most relevant architectural characteristics.

In our algorithm, we have 2 sources of learning. First, the R_{Φ}^{y} and R_{Φ}^{x} that build the posterior distribution from the training set, as explained. This is referred to as learning from seen data in the ground truth. The second source is the test image itself. This is achieved by RBMs R_{Ψ}^{y} and R_{Ψ}^{x}. We utilize the same machine architectures and the network parameters are initialized in the same manner. The only difference in the training is, the sets of directional scanlines are taken from a single layout validated test image, as will be shown. Learning from unseen data, somehow follows the way humans transfer knowledge from intact regions of the scene to parts of less quality. They try to interpret scenes based on past experience, while trying to warp prior perception to fit the scene at hand. We would like to point out, the R_{Ψ}^{y} and R_{Ψ}^{x} are an obvious usage of the denoising property of the RBMs. They are able to construct the distribution upon the scanline parts that conform with the majority, while filtering out the outliers. This has been possible due to the existence of highly repetitive patterns in facades. Of course, this would not be possible without the accurate predictions of the CNN.

Inference. The algorithm is presented in 1. At test time, we perform reconstructions of the CNN output on the trained RBMs. I^0 is subjected to a resizing into s_y and s_x dimensions by nearest neighbour interpolation. The resulting I_y^0 and I_x^0 images are divided into scanlines and the scanlines are binarized, as explained for G_y and G_x. These scanlines aggregated mini-vectors are clamped to

the visible nodes of R^y_Φ and R^x_Φ. The reconstructions $\Gamma^\Phi_{I^0_k}$ in each direction k constitute of posteriors that reflect the visible node tendency to fire. $\Gamma^\Phi_{I^0_k}$ ($k \in \{y, x\}$) are then put in a $r \times c \times M$ matrix form with the original image dimensions to allow merging the 2 directions (horizontal and vertical) into a single vector of classification scores $o^n_\mathcal{L}$ per pixel. The posteriors can easily be mapped to crisp predictions through a meta-learner (explained shortly) to get the multinomial maps I^1. For the n^{th} pixel, the meta-learner produces posteriors $o^n_\mathcal{L}$ and the assigned label becomes

$$\bar{d}_n = \underset{L_m \in \mathcal{L}}{\arg} \max o^n_\mathcal{L} \tag{8.6}$$

I^1 is obtained by concatenating predictions $\bar{d}_n$ for all pixels. I^1 is resized again to fit each s_k ($k \in \{y, x\}$) and converted to binarized scanlines $\Omega_{I^1_y}$ and $\Omega_{I^1_x}$. These are passed on to the next phase. In this way, the algorithm suppresses the pixel labels that are not structurally sound based on past layout experience. We perform a merging step on $\Gamma^\Phi_{I^0_k}$ before relaying them to the next phase because it is understandable that better classifications are obtained when taking both directions into consideration. This is similar to the reasoning found in solving crosswords, in which horizontal and vertical directions can not be considered in isolation.

Algorithm 2 Augmentation procedure

Require: $\Omega_{I^1_y}, \Omega_{I^1_x}, \boldsymbol{\mathcal{L}}$
  $\hat{\Omega}_{I^1_y} = \{\}$
  $\hat{\Omega}_{I^1_x} = \{\}$
  $dim_x \leftarrow countrows(\Omega^2_{I_x})$
  $dim_y \leftarrow countcols(\Omega^2_{I_y})$
  for each $L_m \in \boldsymbol{\mathcal{L}}$ **do**
    for each $k \in \{y, x\}$ **do**
      $S \leftarrow getScanlinesofLabel(\Omega_{I^1_k}, L_m)$
      $t \leftarrow floor(dim_k / \|S\|)$
      $\hat{S} \leftarrow repeat(S, t)$
      $\hat{\Omega}_{I^1_k} \leftarrow append(\hat{\Omega}_{I^1_k}, \hat{S})$
    end for
  end for

Adaptation. R^y_Φ and R^x_Φ are used to enforce architectural guidelines. However, in the aforementioned procedure a rarely encountered facade scanline, but one that is completely valid, is not guaranteed to have a low energy. This is due to the fact the RBM will place the probability mass on the frequently encountered scanlines. It will always produce the models that resemble what has been seen in its training set with sub-optimal adaptability to what is currently visualized in the image. Thus, we need an approach that increases the likelihood of certain labeled scanlines because of their abundance in the test image, regardless of their low frequency in the training ones. Hence, we train RBMs R^y_Ψ and R^x_Ψ on the validated test image scanlines. This time the RBMs will place the probability mass on what is frequent in the current image. In effect, we are extending the receptive field beyond the rows and columns to which the pixel directly belongs and propagating true labels from one area to another.

$\Omega_{I^1_k}$ are now augmented (see below) to form the training visible vectors $\hat{\Omega}_{I^1_k}$ for R^y_Ψ and R^x_Ψ. After learning the parameters of the RBMs, they are applied on the original scanlines obtained from the CNN output I^0_k to produce directional posterior scores $\Gamma^\Psi_{I^0_k}$, which are then merged into a unified scoring map. Experimentally, we found that utilizing I^0_k is better thanI^1_k as the source of scanlines. Logically, I^0_k is more adherent to what is visually perceived in the scene, while I^1_k deviates from the scene at hand to what is structurally sound. In effect, the CNN output is used to launch the refinement of a specialized *s*tructural mind (R^k_Ψ in our case), which is used to revisit the CNN output itself at the end of the pipeline rather than updated versions of it as commonly seen in the literature. The scoring map is interpolated bilinearly to fit it in the r and c dimensions and the final label is chosen again according to the rationale of equation 8.6.

To sum up, the inference is dependent upon the correlation between the hypothesis perceived by appearance and the one suggested by common architectural layouts. That is to say, if a label is found at a certain pixel in I^0 and simultaneously in the set of putative structural labels, then it has a high chance of being assigned to the pixel. Otherwise, the most likely class will be propagated to it, based on the mass of scanlines that have a similar overall configuration in the test image.

In our pipeline, each module relays to its successor a binarized MAP of the classification rather than the absolute posteriors of the labels. This was found to result in better performance measures.

This is an empirical finding for which we can provide a logical justification. For the CNN output, when the true class is missed, there is no guarantee that its posterior will reflect a higher degree of membership of the pixel to the true class than to any other erroneous one. Thus, we made a choice to carry out non-maxima suppression for the pixel appearance posteriors and make the decision entirely on layout basis. As for training the RBMs R^k_Ψ in the adaptation phase, it is widely accepted that training on binary values lead to more efficient training and faster convergence.

Merging posteriors from directional RBMs. A per-pixel decision can be taken by choosing the label with the maximum (maximum average) of the posteriors of both directions. However, we found a more sophisticated method based on higher-level reasoning on the RBM results for merging directional RBM outputs that boosted accuracies. After the training on R^k_Φ was done, we obtained the reconstructions of the training fold on R^k_Φ itself. A meta-feature vector was constructed per-pixel from the reconstructed posteriors in both directions, such that its length is $2 \cdot M$. The target classes are the true pixel labels obtained from the ground truth. We train B a backpropagation Multi-Layer Perceptron (MLP) with single hidden layer, on this data after massive downsampling. To deduce the final labeling, at inference on the test fold, the meta-feature per-pixel is once again constructed but this time from the reconstructions on R^k_Ψ and ran on B. The motivation to using meta-learners in machine learning can be traced back to the stacked generalization algorithm [234], in which predictions made made by base classifiers are aggregated and fed to a communal classifier. It is recommended to use the posteriors in contrast to votes in ensemble learning.

$\Omega_{I^1_k}$ **augmentation.** Imbalanced data is a widely recognized problem for classification techniques [235]. It makes a minority class more prone to be misclassified, due to its relative scarcity in the training set. In our experiments, we noticed that this was highly likely to affect small structures (such as door and chimney) when reconstructing the scanlines on R^y_Ψ and R^x_Ψ (figure 8.2). This is due to the fact a dataset of scanlines built from a single image would have such structures in extremely low counts. Interestingly, this did not occur when the scanlines were tested on R^y_Φ and R^x_Φ, despite being minority classes with the same ratio as in the test image. We realized that the representation power of

a class is not only dependent on the relative count of its instances in the set, but can also be attributed to the size of the class in absolute terms. This phenomenon has been called the *lack of density* and its implication is explained in [236]. The authors state that the minority class in this case is considered as noise and subsequently it is filtered out when building a reliable classification model. We believe the overfitting problem in very small training sets aggravates the imbalance, such that the machine has limited generalization ability not only beyond its training set, but even beyond the majority models in its training set.

We carry out an arbitrary augmentation procedure explained in algorithm 2. It replicates the set of scanlines of each label several times proportional to the ratio of the original count of the set to the size of the image. In this way rare classes will be made more frequent. The procedure leads to increasing the count of small classes and achieving more balanced class-to-class ratios. However, it did not by any means accomplish priors equalization, because adding scanlines for one class inevitably increases other classes as well. Overfitting is an issue when learning from a single image. However, the objective in the first place is not boosting the generalization ability of the image-specific RBM as it will not be applied to unseen data but only reapplied on its training data. As such, we are exploiting the denoising ability of the RBM to conform outlier scanlines to the majority. Learning on RBMs proceeds in batches with an update of the set of parameters after the processing of each batch has ended. Scanlines with minority classes need to be represented in each batch inorder to prevent them from being filtered out as outliers and not contributing to the built conditional probability distribution of $\mathbf{v}$ and $\mathbf{h}$. For this reason, we need an augmentation phase such that scanlines with minority classes are cloned and added to the training set.

8.3 Evaluation

We tested our proposed algorithm on the *ECP-Monge* dataset [224] and the *CMP* dataset [1] as in chapter 6. The *ECP-Monge* contains 104 images of facades in Hausmannian style. There are 8 structures specified in the groundtruth maps. We use the corrected ground truth [57]. The set of

possible labels $\mathcal{L}$ is $\{window, door, shop, wall, roof, balcony, sky, chimney\}$.

The *CMP* dataset is considered more challenging as it contains 378 samples with 12 structures from various (often difficult to model) styles. The set of possible labels $\mathcal{L}$ is

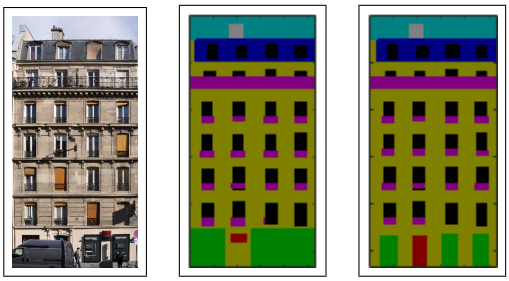
$\{background, facade, sill, balcony, door, blind, molding, deco, shop, window, cornice, pillar\}$.

We would like to point out, manually labeling regions to form ground truth is a highly subjective process. We can see in figure 8.3, 2 valid ground truth maps for the same facade, that are quite different. The ambiguity arises due to the conflict between appearance and common layout, even for humans. We noticed inconsistencies in the labeling in both datasets.Thus, performance measures calculated against these ground truth images should be taken with a grain of salt. And, eye inspection of the labeling quality should be considered.

For the CNN, we retrained the VGG-model on each dataset for 250 epochs (Please refer to figure 8.4), while maintaining the original parameters of learning rate, momentum and weight decay of the pre-trained net. In the experiments, s_y and s_x were unified for all images and set to 250 and 200, respectively. Thus, the number of visible nodes for RBM R^k is $s_k \times M$. In all settings of RBMs, R^k_Φ and R^k_Ψ, the number of hidden nodes was set equal to the number of visible ones. Also, we trained all RBMs for 50 epochs. Our formulation based on the most stable layout representative, the scanline, allowed the RBMs to converge within this unprecedented small number of learning epochs (figure 8.5). The MLP B was trained on a maximum of 8% of the nimber of pixels available in the training fold. The training algorithm was Scaled Conjugate Gradient (SGD) [237]. Training was stopped when the gradient drops to a value of $1.00\,e-06$. For the *ECP-Monge* dataset, the number of nodes in hidden layer was 15 and the number was 23 for the *CMP* dataset.

As a performance measure, we utilize the pixel accuracy. It is calculated as $TP/\left(TP+FN\right)$. True Positives TP and False negatives FN are determined overall the set of image pixels. We report our results based on 5-fold cross validation to ensure fair comparison between our algorithm Deep Facade Parsing (DPF-Ψ) and related work. We present the results in table 8.2.

[221] is the highest reported accuracy on the *ECP-Monge* dataset, to-date. It achieves 90.0% in the phase based on image features (equivalent to our appearance-based module) and 91.4% after



(a) (b) (c)

Figure 8.3: (a) A sample image. (b) First possible ground truth map. (c) Second possible ground truth map.

structural improvement. Their results are reported for the older version of ground truth images based on 7 classes. The disregarded class is for the chimney structure. This kind of minority classes is often a bottleneck for the algorithms. In the literature, the lowest class accuracies were seen for doors and chimneys. Even in their own reported results, the door was the class of lowest accuracy. Thus, one can not be sure what the true overall accuracy will be, if the chimney structure was included as in our case. Same reasoning applies for [112].

The results show that we are on-par with state-of-the-art algorithms in terms of accuracy. More importantly, our algorithm highlights the benefit of context in image analysis. We report one of the highest accuracy gains, after inclusion of layout cues, defined as $A_2 - A_1$, without the dataset tailored refining rules found in [156, 58]. For the *CMP* dataset, it is expected that any appearance-based improvements will synergisticaly boost the structure module to even higher accuracy figures. Other computer vision research has noticed an increase in accuracy when utilizing VGG-19 or ResNet CNN models, in place of VGG-16. In figures 8.6, 8.7 and 8.8, we display results of a selection of samples.

As a further investigation of how this algorithm compares to the TRW-S optimized formulation presented in chapter 7, we provide the standard deviation of the results calculated based on the accuracies of the images in relation to the overall dataset average in table 8.1. This provides an indication of the stability of the performance of the algorithms. The smaller the value, the more stable the performance. Clearly, from the final accuracy figures, we can declare the deep learning pipeline as the

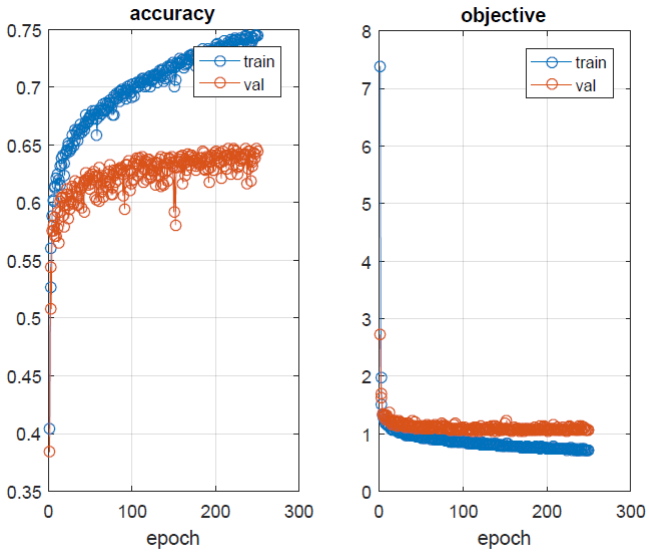


Figure 8.4: A sample graph for the accuracy and objective figures over the training epochs of the CNN. Accuracy is calculated per-pixel against ground truth. The objective is a cost function calculated as the cross entropy between the ground truth and the predicted distribution.

Dataset	*ECP-Monge*	*CMP*
IASC	3.87	8.76
DPF-Φ	2.63	9.83

Table 8.1: Standard deviation of accuracies of approaches presented in chapters 7 and 8 for comparison purposes.

higher performing approach. But, one can not neglect the fact that this is partially attributed to the superiority of the CNN in the appearance module. TRW-S optimized layout module achieves a higher gain based on structural cues. It was able to recover the correct topology even when presented with severly corrupted segmentations resulting from the RF-based appearance module. However, the fact that the DPF-Ψ requires no parameter setting renders it the more promising approach.

As a self-test, we examine 2 variants of our algorithm to evaluate the different aspects proposed in its pipeline. These are:

- *variant 1*- Same as DFP with the per-pixel classification obtained through maximizing the posterior based on the average of both directions to get the labeling. This is used to assess the

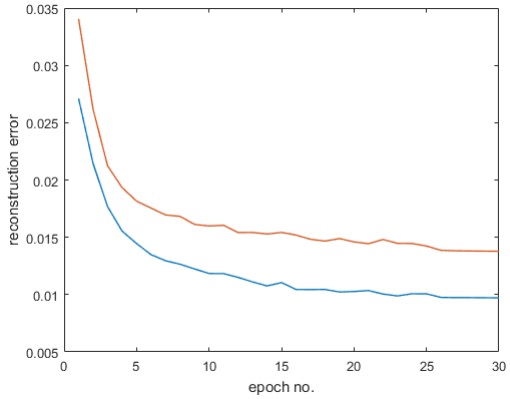


Figure 8.5: A sample graph for the reconstruction error on the validation set over the training epochs. R^y_Φ consistently has a higher error than R^x_Φ, due to the higher variability of vertical scanlines over horizontal ones.

Dataset	*ECP-Monge*		*CMP*	
Method	A_1	A_2	A_1	A_2
[1]	59.60	84.20	33.20	60.30
[58]	84.75$*$	88.02$*$	-	-
[156]	86.71	90.34	-	-
[112]	90.10$*$	91.30$*$	-	-
[221]	90.80$*$	91.40$*$	66.20	68.10
DPF-Ψ	86.45	91.31	61.46	69.02

Table 8.2: Overall pixel accuracies based on appearance cues A_1 and when combined with layout cues A_2. The references marked with $*$ are included for completeness of results and are not suitable for direct comparison.

goodness of the MLP as a merging criterion. For the *ECP-Monge* dataset the pixel accuracy was 90.87 and for the *CMP* dataset the figure was 67.70.

- *variant 2*- Running the test image on R^y_Φ and R^x_Φ only, without the adaptation phase and its complementary augmentation. The results for *ECP-Monge* dataset and the *CMP* dataset were 90.49 and 65.54, respectively.

There is a recent tendency to evaluate performance based on the Intersection-over-Union measure (IoU), popularized by Pascal VOC competition [77]. We include also a more recent evaluating crite-

Method	[221]∗	DPF-Ψ	[58]∗
window	82.00	84.21	78.00
door	81.00	70.25	71.00
shop	93.00	92.71	95.00
wall	93.00	94.04	89.00
roof	98	90.45	79.00
balcony	89.00	88.98	87.00
sky	98.00	95.67	96.00
chimney	N/C	79.98	N/C
A_3	89.5	87.04	85.22
IoU	80.5	77.95	-

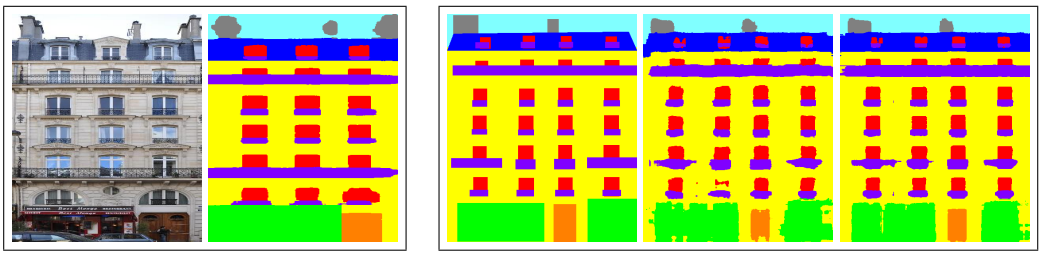
Table 8.3: Per-class pixel accuracy, A_3 (Average class pixel accuracy), and IoU results on the *ECP-Monge* dataset. N/C stands for Not Considered.

rion which is the IoU measure. It is defined as $TP/\left(TP+FN+FP\right)$. True Positives TP, False Negatives FN and False Positives FP are calculated per class and then averaged as reported in [120]. Also, we include a detailed per class assessment based on pixel accuracy. The results are shown in tables 8.3 and 8.4 for the datasets. A point to note is that we attained a higher mean class accuracy than [221], but a lower mean IoU figure on the *CMP* dataset. This was accentuated in classes of small size in both datasets. This is due to the fact that classification errors affect more drastically classes of lower count. Therefore, the introduction of FP in IoU caused the figures to drop more acutely, provided that the algorithm is not biased against small-sized classes in the first place. Otherwise, TP will also drop and the class accuracies will no longer be higher than that of [221]. The invariance of our algorithm towards size is a by-product of the augmentation process and is proved through the classes accuracies.

The value of the adaptation was more evident for the *CMP* dataset. Because, *ECP-Monge* manifest much more stable arrangements among seen and unseen samples that are efficiently learned during training. Thus, there is no much need for customization of the layout. In contrast, *CMP* which relies on adaptation to be able to generalize efficiently.

Method	[221]	DPF-Ψ
background	-	36.02
facade	-	81.29
sill	-	74.12
balcony	-	47.41
door	-	65.98
blind	-	68.89
molding	-	74.24
deco	-	70.43
shop	-	48.26
window	-	61.82
cornice	-	54.42
pillar	-	60.32
A_3	48.9	61.93
IoU	37.5	30.51

Table 8.4: Per-class pixel accuracy, A_3 (Average class pixel accuracy), and IoU results on the *CMP* dataset.



(a) (b)

Figure 8.6: (a) A sample image (left) and the result of DPF-Ψ (right). The output of one of the rounded windows shows that the algorithm can handle *to some extent* the case of architecural structures with rounded boundaries without enforcing right angles and straight lines. (b) A sample of groundtruth (left), DPF-Φ (middle) and DPF-Ψ (right) results on the image. It shows that training on the image was able to recover the missing window because the correct label was propagated to it from similar scanlines.

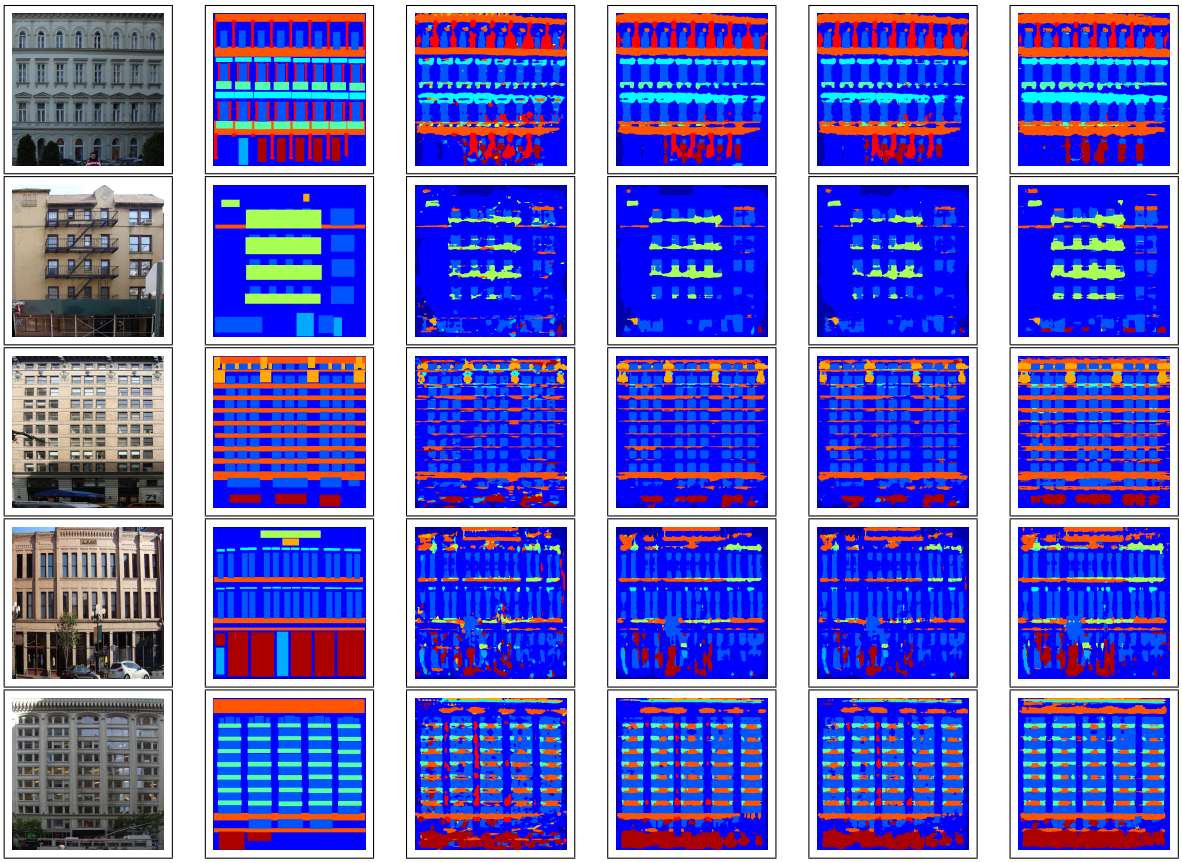


Figure 8.7: (a) A sample image. (b) Ground truth map. (c) CNN output I^0. (d) Variant 1. (e) Variant 2. (f) DPF-Ψ.

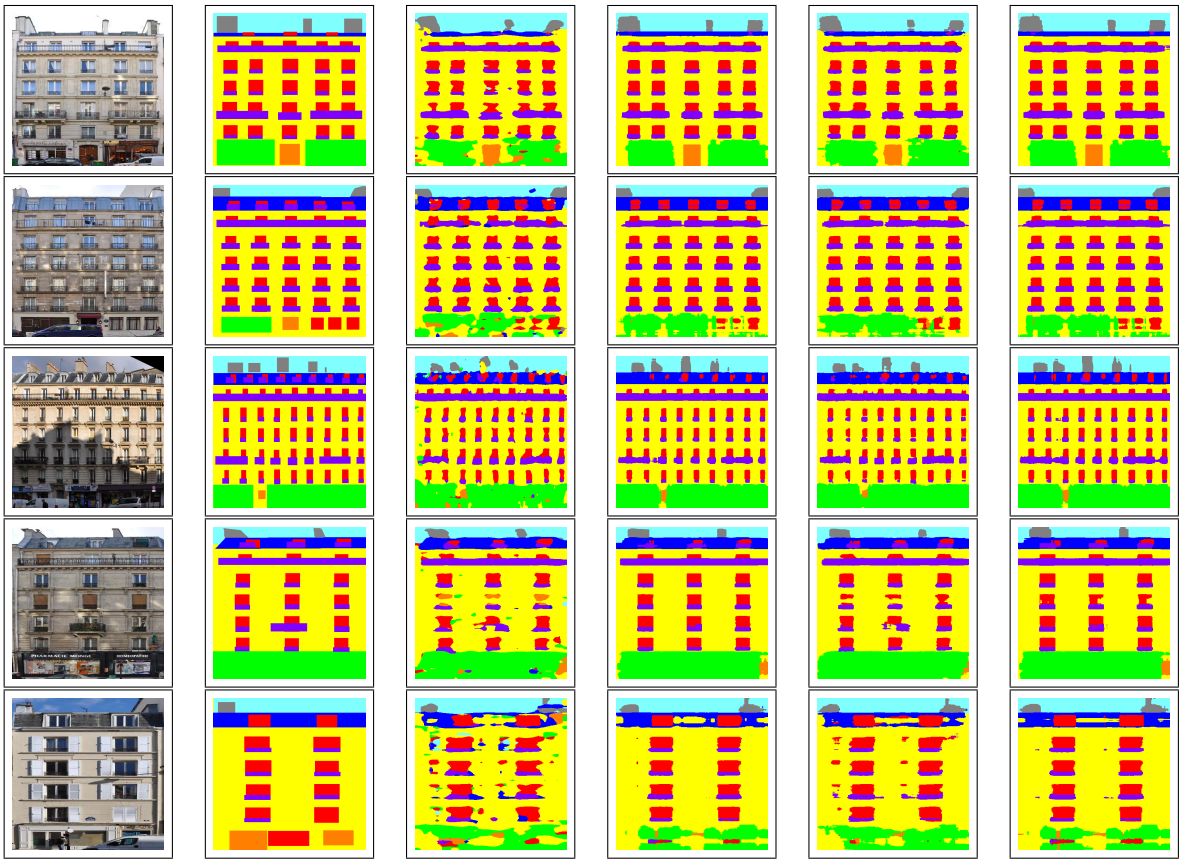


Figure 8.8: (a) A sample image. (b) Ground truth map. (c) CNN output I^0. (d) Variant 1. (e) Variant 2. (f) DPF-Ψ.

8.4 Conclusion

We have presented a pipeline for facade parsing. It relies on the state-of-the-art techniques of computer vision, and acheives on-par results with related research efforts. We do not include any ad hoc post-processing steps, nor do we manually specify any architecture-based features. The pipeline is initialized with classifications from the VGG-16 convolution model customized to semantic segmentation. The results are further improved through a probabilistic shape prior captured by trained RBMs. We present a novel idea to learn from test images, to increase the generalization ability of the algorithm. We illustrate the importance of dataset augmentation for severely small imbalanced datasets, resulting from a single test image. This work is in [27].

Chapter 9

Discussion and Conclusion

In this dissertation, we presented algorithms for model fitting in the area of visual perception. The two problems tackled are: realizing semantic interpretation models for building facades and fitting of primitive geometric models to data points defined sparsely on irregular lattices. In all applications, we exploit domain specific layout priors. Throughout the work, we maintained several design principles.

- We are keen on modeling the interaction between the primitives of the problem (3d points, pixels and superpixels) such that when a decision is made its effect is propagated back to the whole of the available data. This was achievable due to our use of graphical models with various connectivity patterns. In contrast, some algorithms make decisions locally for each image primitive in isolation while others carry out interaction modeling on whole region level without the ability to fine tune at the primitive level. Our approach achieves concurrency between two modules independently handled in most pattern recognition systems: segmentation and recognition. Usually, segmentation precedes and relays to the recognition module whole blobs are to be classified. This is a sub-optimal approach. There is evidence from the neuroscience field [238, 239] that humans use recognition cues to enhance the partitioning of images into objects. This is also verified by our competitive results.

- We provide more elegant automated solutions to the aforementioned problems, that are not

dataset-dependent, we eliminate the use of handwritten priors and manually specified thresholds. In worst case scenario when parameter setting is inevitable, it is automatically learned through established optimization techniques or dynamically derived.

- We try to enrich our formulations with the largest number of priors statistically derived to fully exploit domain knowledge. And, we were able to keep the formulation tractable so that optimization techniques can effectively handle it.

- We make heavy use of meta-features which relied on absolute distances to classes, ranked memberships and class posteriors. The experimental evaluation proved that these meta-features were superior to normal feature vectors.

9.1 Summary

We give a review of semantic segmentation in generic scenes in chapter 2 with a special focus on the work that uses context information either on the object or global level. In addition, we introduce the basics of CNNs and investigate its state-of-the-art architectures commonly exploited in similar applications. In chapter 3, we get more specific to our application and we present a thorough survey of related work in the field of facade analysis while categorizing the efforts. We present the needed background for understanding subsequent sections in chapter 4, where we detail optimization techniques defined on random fields and give the formulation of RBMs with their application to segmentation.

In the domain of geometric multi-model where RANSAC and energy-based algorithms prevail, we were able to make a contribution based on guided sampling. Chapter 5 presents a novel algorithm that relies on MST to aggregate data points from which a model is hypothesized. We introduce a stable alignment and margin of error criteria that allowed the algorithm to backtrack to the best model built on the locality of points. We make use of the idea of multiplicity to support the election of the final set of models by assuming that erroneous models resulting from outliers are not surrounded by alikes. For this reason, we quantify similarity between models by a measure that improves on the standard measure of Jaccuard distance.

The main ideas of the algorithm presented in chapter 6 include the use of watershed superpixels which downsized the computation burden by several folds while reducing the effect of errors in the segmentation module if regions of larger size were propagated forward. Also, the use of meta-feature vectors compared favourably to plain feature vectors. More importantly, the combination of appearance information with layout information in a single framework solved effectively by the TRW-S tool. The layout information included properties such as spatial coherence, approximate structural location, structure ordering, recurring structural adjacencies and translational symmetry.

In chapter 7, We use RBMs to denoise CNN output for facade parsing based on the VGG architecture adapted to segmentation. The multi-class label in each pixel is converted to one-hot vector, which are then stacked along each vertical or horizontal line to form binary patterns as input of two sets of RBMs. One pair of such RBMs are learned in the training set with ground truth labels and other pair of RBMs are learned on-the-fly with test data to allow some knowledge transfer from regions of high-quality labels to noisy ones. Our graph-based formulation with the full connectivity between the pixels and their latent representation allows modeling of the intrinsic dependency relations subsequently allowing synergistic inference among pixels. Interest in topological context is on the rise, motivated by recent studies which show the continued superiority of man over machinery in using it for visual perception. Formulations such as ours will help boost the RBM as a powerful tool in structural modeling beyond single object of focus.

Truly, the TRW-S layout optimization of the proposed algorithm of chapter 6 achieves the higher accuracy gain, due to incorporation of structure cues, over the deep learning pipeline in chapter 8. However, the fact that deep learning techniques are fully automated, without the need for fine tuning even of hyper-parameters, makes it the more appealing option for solving computer vision problems. Heavy machinery is utilized in the pipeline of chapter 8, which incurs a high computational cost, but this drawback can be compensated by the increased capabilities of computers on the hardware front.

9.2 Future extensions

There are known limitations of the work that could be further investigated. Truly, rectification is a standard preprocessing step in most image analysis pipelines, but it would be interesting to see how the proposed algorithms will behave with significant orientations (or rotations) of the image. For algorithm in chapter 6, unconstraining the translations to x and y directions and collecting priors from tilted examples will make the algorithm applicable to this type of images. At first glance, the work of the neural network pipeline seems that its heavy reliance on the vertical and horizontal scanlines will render it not suitable for such domain. However, as we have seen in the output samples, the pipeline was lenient with the straight boundaries and right angles assumption that prevents approaches of rectangular bounding boxes from being applied to non-rectified images.

Another interesting area is segmenting facade images with occlusions. The challenge in the literature is encountered in 2 settings. Either the occlusion is manually specified by a user (the black box problem) or it is unspecified and the occluding object is a natural part of the scene. In all cases, RBMs are acclaimed [228, 240] for their ability to generate true structure of image even in the presence of a high percentage of missing or erroneous data.

We would also like to experiment with deeper architectures of DBM and/or DBN on the structural front. A thing that is expected to boost the latent representation of the layout on the hidden nodes to even higher-level, more abstract form that leads to better generalization. But, the added time and complexity should be satisfactorily paid-off with substantial increased accuracy. Another possibility is building a spatial hierarchy of RBMs, where RBMS at the lowest level are clamped to the scanlines and the ones at higher levels are fed with merged outputs from neighboring RBMs of the lower level. In this way, the whole of the image could be perceived at the top-most layers of the architecture.

An extension of the work presented in chapters 7 and 8 is applying the algorithms on point clouds. This involves the incorporation of depth information and will be used in the construction of fully textured and recognized 3d models of buildings. A recent survey [241] evaluating deep learning techniques in 3d vision, concludes that so far applying the techniques has been more successful in the

2d domain. This makes improvements in the 3d domain an interesting point of research.

Last but not least, is the application of the semantic segmentation algorithms in generic settings where there are no alignment between structures in the layout and there are articulated objects. However, most objects in scenes have a locality prior, vertical and horizontal arrangement order and inherent affinity to image zones. These characteristics would definitely benefit from the RBM imposed priors. However, the increased variability might necessitate the addition of underlying layers prior to RBMs to achieve invariance.

Bibliography

[1] R. Tyleček and R. Šára, *Spatial Pattern Templates for Recognition of Objects with Regular Structure*, ch. Spatial Pa, pp. 364–374. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[2] R. Mottaghi, S. Fidler, J. Yao, R. Urtasun, and D. Parikh, "Analyzing Semantic Segmentation Using Hybrid Human-Machine CRFs," in *2013 {IEEE} Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pp. 3143–3150, {IEEE} Computer Society, 2013.

[3] J. Wagemans, J. Feldman, S. Gepshtein, R. Kimchi, J. R. Pomerantz, P. a. van der Helm, and C. van Leeuwen, "A century of Gestalt psychology in visual perception: II. Conceptual and theoretical foundations.," *Psychological bulletin*, vol. 138, pp. 1218–52, Nov. 2012.

[4] D. Mumford, "Neuronal Architectures for Pattern-theoretic Problems," in *Large-Scale Theories of the Cortex*, pp. 125–152, MIT Press, 1994.

[5] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object Detection with Discriminatively Trained Part-Based Models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, pp. 1627–1645, Sept. 2010.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[7] D. K. G. Heitz, "Learning Spatial Context: Using Stuff to Find Things," in *ECCV*, 2008.

[8] K. Chellapilla, S. Puri, and P. Simard, "High Performance Convolutional Neural Networks for Document Processing," in *Tenth International Workshop on Frontiers in Handwriting Recognition* (G. Lorette, ed.), (La Baule (France)), Universit{é} de Rennes 1, Suvisoft, Oct. 2006.

[9] K. Dang and J. Yuan, "Location Constrained Pixel Classifiers for Image Parsing with Regular Spatial Layout," in *Proceedings of the British Machine Vision Conference*, BMVA Press, 2014.

[10] J.-M. Marin, K. L. Mengersen, and C. Robert, "Bayesian modelling and inference on mixtures of distributions," in *Handbook of Statistics: Volume 25* (D. Dey and C. R. Rao, eds.), Elsevier, 2005.

[11] A. Dame, V. A. Prisacariu, C. Y. Ren, and I. D. Reid, "Dense Reconstruction Using 3D Object Shape Priors," in *2013 {IEEE} Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pp. 1288–1295, 2013.

[12] I. T. Jolliffe, *Principal Component Analysis*. Springer Verlag, 1986.

[13] V. Mnih and G. E. Hinton, *Learning to Detect Roads in High-Resolution Aerial Images*, pp. 210–223. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems*, NIPS'12, (USA), pp. 1097–1105, Curran Associates Inc., 2012.

[15] O. L. Mangasarian and D. R. Musicant, "Lagrangian Support Vector Machines," *J. Mach. Learn. Res.*, vol. 1, pp. 161–177, Sept. 2001.

[16] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, pp. 504–507, July 2006.

[17] E. Borenstein, E. Sharon, and S. Ullman, "Combining Top-Down and Bottom-Up Segmentation," in *Proceedings of the 2004 Conference on Computer Vision and Pattern Recogni-*

tion Workshop (CVPRW'04) Volume 4 - Volume 04, CVPRW '04, (Washington, DC, USA), pp. 46—-, IEEE Computer Society, 2004.

[18] A. Levin and Y. Weiss, "Learning to Combine Bottom-Up and Top-Down Segmentation," *Int. J. Comput. Vision*, vol. 81, pp. 105–118, Jan. 2009.

[19] B. Hariharan, P. Arbelaez, R. B. Girshick, and J. Malik, "Simultaneous Detection and Segmentation," *CoRR*, vol. abs/1407.1, 2014.

[20] R. Szeliski, *Computer Vision: Algorithms and Applications*. New York, NY, USA: Springer-Verlag New York, Inc., 1st ed., 2010.

[21] R. Chellappa, "Mathematical Statistics and Computer Vision," *Image Vision Comput.*, vol. 30, pp. 467–468, Aug. 2012.

[22] R. Fathalla and G. Vogiatzis, "Multi-model fitting based on Minimum Spanning Tree," in *25th British Machine Vision Conference, Nottingham, United Kingdom, 1-5 September*, 2014.

[23] V. Kolmogorov, "Convergent Tree-Reweighted Message Passing for Energy Minimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 1568–1583, Oct. 2006.

[24] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[25] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.

[26] R. Fathalla and G. Vogiatzis, "Optimization of Facade Segmentation Based on Layout Priors," in *Computer Analysis of Images and Patterns, CAIP*, (Ystad, Sweden), Springer Verlag's series Lecture Notes in Computer Science (LNCS), 2017.

[27] R. Fathalla and G. Vogiatzis, "A Deep Learning Pipeline for Semantic Facade Segmentation," in *28th British Machine Vision Conference, London, United Kingdom, 4-7 September*, 2017.

[28] L. Vincent and P. Soille, "Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, pp. 583–598, June 1991.

[29] D. Comaniciu and P. Meer, "Mean Shift : A Robust Approach Toward Feature Space Analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, 2002.

[30] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *INTERNATIONAL JOURNAL OF COMPUTER VISION*, vol. 1, no. 4, pp. 321–331, 1988.

[31] Y.-T. Chen and D.-C. Tseng, "Medical Image Segmentation Based on the Bayesian Level Set Method.," in *MIMI* (X. W. Gao, H. Müller, M. Loomes, R. Comley, and S. Luo, eds.), vol. 4987 of *Lecture Notes in Computer Science*, pp. 25–34, Springer, 2007.

[32] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation.," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, 2000.

[33] Y. Boykov and G. Funka-Lea, "Graph Cuts and Efficient N-D Image Segmentation," *Int. J. Comput. Vision*, vol. 70, pp. 109–131, Nov. 2006.

[34] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior Recognition via Sparse Spatio-temporal Features," in *Proceedings of the 14th International Conference on Computer Communications and Networks*, ICCCN '05, (Washington, DC, USA), pp. 65–72, IEEE Computer Society, 2005.

[35] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, (Washington, DC, USA), pp. 886–893, IEEE Computer Society, 2005.

[36] T. Wang and H. Snoussi, "Histograms of Optical Flow Orientation for Visual Abnormal Events Detection.," in *AVSS*, pp. 13–18, IEEE Computer Society, 2012.

[37] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-Up Robust Features (SURF)," *Comput. Vis. Image Underst.*, vol. 110, pp. 346–359, June 2008.

[38] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int. J. Comput. Vision*, vol. 60, pp. 91–110, Nov. 2004.

[39] J. Yang, B. Price, S. Cohen, and M.-H. Yang, "Context Driven Scene Parsing with Attention to Rare Classes," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '14, (Washington, DC, USA), pp. 3294–3301, IEEE Computer Society, 2014.

[40] J. Tighe, M. Niethammer, and S. Lazebnik, "Scene Parsing with Object Instances and Occlusion Ordering," in *2014 {IEEE} Conference on Computer Vision and Pattern Recognition, {CVPR} 2014, Columbus, OH, USA, June 23-28, 2014*, pp. 3748–3755, {IEEE} Computer Society, 2014.

[41] Y. Wu, "Object Retrieval and Localization with Spatially-constrained Similarity Measure and k-NN Re-ranking," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, (Washington, DC, USA), pp. 3013–3020, IEEE Computer Society, 2012.

[42] M. George, "Image Parsing with a Wide Range of Classes and Scene-Level Context," *CoRR*, vol. abs/1510.07136, 2015.

[43] M. Collins, R. E. Schapire, and Y. Singer, "Logistic Regression, AdaBoost and Bregman Distances," *Mach. Learn.*, vol. 48, pp. 253–285, Sept. 2002.

[44] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "SLIC Superpixels Compared to State-of-the-Art Superpixel Methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, pp. 2274–2282, Nov. 2012.

[45] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, pp. 971–987, July 2002.

[46] T. Leung and J. Malik, "Representing and Recognizing the Visual Appearance of Materials Using Three-dimensional Textons," *Int. J. Comput. Vision*, vol. 43, pp. 29–44, June 2001.

[47] T. Heskes, "Bias/Variance Decompositions for Likelihood-Based Estimators.," *Neural Computation*, vol. 10, no. 6, pp. 1425–1433, 1998.

[48] Y. Ren, C. Chen, S. Li, and C.-C. J. Kuo, "{GAL:} {A} global-attributes assisted labeling system for outdoor scenes," *J. Visual Communication and Image Representation*, vol. 42, pp. 192–206, 2017.

[49] D. Hoiem, A. A. Efros, and M. Hebert, "Closing the loop in scene interpretation," in *2008 {IEEE} Computer Society Conference on Computer Vision and Pattern Recognition {(CVPR} 2008), 24-26 June 2008, Anchorage, Alaska, {USA}*, {IEEE} Computer Society, 2008.

[50] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient Graph-Based Image Segmentation," *Int. J. Comput. Vision*, vol. 59, pp. 167–181, Sept. 2004.

[51] X. Fu, C.-Y. Wang, C. Chen, C. Wang, and C.-C. J. Kuo, "Robust Image Segmentation Using Contour-Guided Color Palettes," in *2015 {IEEE} International Conference on Computer Vision, {ICCV} 2015, Santiago, Chile, December 7-13, 2015*, pp. 1618–1625, 2015.

[52] W. Huang and X. Gong, "Fusion Based Holistic Road Scene Understanding," *CoRR*, vol. abs/1406.7, 2014.

[53] J. Liu and X. Gong, "Guided Depth Enhancement via Anisotropic Diffusion.," in *PCM* (B. Huet, C.-W. Ngo, J. Tang, Z.-H. Zhou, A. G. Hauptmann, and S. Yan, eds.), vol. 8294 of *Lecture Notes in Computer Science*, pp. 408–417, Springer, 2013.

[54] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Magazine Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[55] R. Socher, B. Huval, B. Bhat, C. D. Manning, and A. Y. Ng, "Convolutional-recursive Deep Learning for 3D Object Classification," in *Proceedings of the 25th International Conference on Neural Information Processing Systems*, NIPS'12, (USA), pp. 656–664, Curran Associates Inc., 2012.

[56] B. Shuai, Z. Zuo, G. Wang, and B. Wang, "Scene Parsing with Integration of Parametric and Non-parametric Models," *CoRR*, vol. abs/1604.0, 2016.

[57] A. Martinović, M. Mathias, J. Weissenberg, and L. Van Gool, "A Three-Layered Approach to Facade Parsing," in *Computer Vision – ECCV 2012* (A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, eds.), vol. 7578 of *Lecture Notes in Computer Science*, pp. 416–429, Springer, 2012.

[58] M. Mathias, A. Martinović, and L. Van Gool, "ATLAS: A Three-Layered Approach to Facade Parsing," *International Journal of Computer Vision*, vol. 118, no. 1, pp. 22–48, 2016.

[59] M. A. Fischler and R. A. Elschlager, "The Representation and Matching of Pictorial Structures," *IEEE Trans. Comput.*, vol. 22, pp. 67–92, Jan. 1973.

[60] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Visual Object Detection with Deformable Part Models," *Commun. ACM*, vol. 56, pp. 97–105, Sept. 2013.

[61] R. Fergus, P. Perona, A. Zisserman, and O. P. U. K, "Object class recognition by unsupervised scale-invariant learning," in *In CVPR*, pp. 264–271, 2003.

[62] A. Vezhnevets and V. Ferrari, "Object localization in ImageNet by looking out of the window," in *Proceedings of the British Machine Vision Conference 2015, {BMVC} 2015, Swansea,*

UK, September 7-10, 2015 (X. Xie, M. W. Jones, and G. K. L. Tam, eds.), pp. 27.1—-27.12, {BMVA} Press, 2015.

[63] S. Manen, M. Guillaumin, and L. V. Gool, "Prime Object Proposals with Randomized Prim's Algorithm," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*, ICCV '13, (Washington, DC, USA), pp. 2536–2543, IEEE Computer Society, 2013.

[64] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[65] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large Margin Methods for Structured and Interdependent Output Variables," *J. Mach. Learn. Res.*, vol. 6, pp. 1453–1484, Dec. 2005.

[66] G. H. Bakir, T. Hofmann, B. Schölkopf, A. J. Smola, B. Taskar, and S. V. N. Vishwanathan, *Predicting Structured Data (Neural Information Processing)*. The MIT Press, 2007.

[67] J. Oramas M. and T. Tuytelaars, "Recovering Hard-to-find Object Instances by Sampling Context-based Object Proposals," *Comput. Vis. Image Underst.*, vol. 152, pp. 118–130, Nov. 2016.

[68] J. R. Uijlings, K. E. Sande, T. Gevers, and A. W. Smeulders, "Selective Search for Object Recognition," *Int. J. Comput. Vision*, vol. 104, pp. 154–171, Sept. 2013.

[69] C. L. Zitnick and P. Dollár, "Edge Boxes: Locating Object Proposals from Edges," in *Computer Vision - {ECCV} 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part {V}* (D. J. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), vol. 8693 of *Lecture Notes in Computer Science*, pp. 391–405, Springer, 2014.

[70] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the Objectness of Image Windows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, pp. 2189–2202, Nov. 2012.

[71] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proceedings of the National Academy of Sciences*, vol. 101, pp. 5228–5235, Apr. 2004.

[72] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[73] D. G. Lowe, "Object Recognition from Local Scale-Invariant Features," in *Proceedings of the International Conference on Computer Vision-Volume 2*, ICCV '99, (Washington, DC, USA), pp. 1150–1156, IEEE Computer Society, 1999.

[74] Y. Amit and A. Kong, "Graphical Templates for Model Registration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, pp. 225–236, Mar. 1996.

[75] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. L. Yuille, "The Role of Context for Object Detection and Semantic Segmentation in the Wild," in *2014 {IEEE} Conference on Computer Vision and Pattern Recognition, {CVPR} 2014, Columbus, OH, USA, June 23-28, 2014*, pp. 891–898, {IEEE} Computer Society, 2014.

[76] J. a. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu, "Free-Form Region Description with Second-Order Pooling," *{IEEE} Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 6, pp. 1177–1189, 2015.

[77] M. Everingham, S. M. A. Eslami, L. Van˜Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes Challenge: A Retrospective," *International Journal of Computer Vision*, vol. 111, pp. 98–136, Jan. 2015.

[78] T. Kadir and M. Brady, "Saliency, Scale and Image Description," *Int. J. Comput. Vision*, vol. 45, pp. 83–105, Nov. 2001.

[79] C. Cadena, A. R. Dick, and I. D. Reid, "A fast, modular scene understanding system using context-aware object detection," in *{IEEE} International Conference on Robotics and Automation, {ICRA} 2015, Seattle, WA, USA, 26-30 May, 2015*, pp. 4859–4866, IEEE, 2015.

[80] L.-J. Li, H. Su, Y. Lim, and L. Fei-Fei, "Object Bank: An Object-Level Image Representation for High-Level Visual Recognition," *Int. J. Comput. Vision*, vol. 107, pp. 20–39, Mar. 2014.

[81] C. D. C. Lerma and J. Kosecka, "Semantic segmentation with heterogeneous sensor coverages," in *2014 {IEEE} International Conference on Robotics and Automation, {ICRA} 2014, Hong Kong, China, May 31 - June 7, 2014*, pp. 2639–2645, IEEE, 2014.

[82] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast Feature Pyramids for Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, pp. 1532–1545, Aug. 2014.

[83] Hamid Izadinia, Fereshteh Sadeghi, and Ali Farhadi, "Incorporating Scene Context and Object Layout into Appearance Modeling," *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 00, pp. 232–239, 2014.

[84] R. B. Girshick, F. N. Iandola, T. Darrell, and J. Malik, "Deformable part models are convolutional neural networks.," in *CVPR*, pp. 437–446, IEEE Computer Society, 2015.

[85] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *CoRR*, vol. abs/1502.0, 2015.

[86] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines.," in *ICML* (J. Fürnkranz and T. Joachims, eds.), pp. 807–814, Omnipress, 2010.

[87] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Comput. Vision*, vol. 115, pp. 211–252, Dec. 2015.

[88] Y. Gal and Z. Ghahramani, "Dropout As a Bayesian Approximation: Representing Model Uncertainty in Deep Learning," in *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pp. 1050–1059, JMLR.org, 2016.

[89] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *CoRR*, vol. abs/1409.1, 2014.

[90] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 {IEEE} Conference on Computer Vision and Pattern Recognition, {CVPR} 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 770–778, {IEEE} Computer Society, 2016.

[91] Z. Wu, C. Shen, and A. van den Hengel, "Wider or Deeper: Revisiting the ResNet Model for Visual Recognition," *CoRR*, vol. abs/1611.1, 2016.

[92] D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "High-Performance Neural Networks for Visual Object Classification," *CoRR*, vol. abs/1102.0, 2011.

[93] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning Hierarchical Features for Scene Labeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, pp. 1915–1929, Aug. 2013.

[94] S. Gupta, R. B. Girshick, P. Arbelaez, and J. Malik, "Learning Rich Features from {RGB-D} Images for Object Detection and Segmentation," *CoRR*, vol. abs/1407.5, 2014.

[95] J. Long, E. Shelhamer, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *CoRR*, vol. abs/1411.4, 2014.

[96] H. Noh, S. Hong, and B. Han, "Learning Deconvolution Network for Semantic Segmentation," *CoRR*, vol. abs/1505.0, 2015.

[97] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," in *Computer Vision - {ECCV} 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part {I}* (D. J. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), vol. 8689 of *Lecture Notes in Computer Science*, pp. 818–833, Springer, 2014.

[98] B. Hariharan, P. A. Arbeláez, R. B. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in {*IEEE*} *Conference on Computer Vision and Pattern*

Recognition, {CVPR} 2015, Boston, MA, USA, June 7-12, 2015, pp. 447–456, {IEEE} Computer Society, 2015.

[99] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks.," *CoRR*, vol. abs/1312.6, 2013.

[100] M. Kampffmeyer, A.-B. r. Salberg, and R. Jenssen, "Semantic Segmentation of Small Objects and Modeling of Uncertainty in Urban Remote Sensing Images Using Deep Convolutional Neural Networks," in *2016 {IEEE} Conference on Computer Vision and Pattern Recognition Workshops, {CVPR} Workshops 2016, Las Vegas, NV, USA, June 26 - July 1, 2016*, pp. 680–688, {IEEE} Computer Society, 2016.

[101] J. Dai, K. He, and J. Sun, "Convolutional feature masking for joint object and stuff segmentation," in *{IEEE} Conference on Computer Vision and Pattern Recognition, {CVPR} 2015, Boston, MA, USA, June 7-12, 2015*, pp. 3992–4000, {IEEE} Computer Society, 2015.

[102] S. G. Mallat and Z. Zhang, "Matching Pursuits with Time-frequency Dictionaries," *Trans. Sig. Proc.*, vol. 41, pp. 3397–3415, Dec. 1993.

[103] A. Newell, K. Yang, and J. Deng, "Stacked Hourglass Networks for Human Pose Estimation," *CoRR*, vol. abs/1603.0, 2016.

[104] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *CoRR*, vol. abs/1606.0, 2016.

[105] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '14, (Washington, DC, USA), pp. 1701–1708, IEEE Computer Society, 2014.

[106] J. García-Gago, D. González-Aguilera, J. Gómez-Lahoz, and J. I. San José-Alonso, "A Photogrammetric and Computer Vision-Based Approach for Automated 3D Architectural Modeling and Its Typological Analysis," *Remote Sensing*, vol. 6, no. 6, pp. 5671–5691, 2014.

[107] J. Hu, S. You, and U. Neumann, "Approaches to large-scale urban modeling," *IEEE Computer Graphics and Applications*, vol. 23, no. 6, pp. 62–69, 2003.

[108] J. Hu, S. You, U. Neumann, and K. K. Park, "Building modeling from LiDAR and aerial imagery," in *ASPRS*, vol. 4, pp. 23–28, 2004.

[109] C. Santagati, L. Inzerillo, and F. D. Paola, "Image-based Modeling Techniques for Architectural Heritage 3D Digitalization: Limits and Potentials," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL-5, pp. 555–560, Sept. 2013.

[110] O. Teboul, I. Kokkinos, L. Simon, P. Koutsourakis, and N. Paragios, "Parsing Facades with Shape Grammars and Reinforcement Learning," {*IEEE*} *Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1744–1756, 2013.

[111] H. Zhang, K. Xu, W. Jiang, J. Lin, D. Cohen-Or, and B. Chen, "Layered Analysis of Irregular Facades via Symmetry Maximization," *ACM Trans. Graph.*, vol. 32, pp. 121:1—-121:13, July 2013.

[112] M. Kozinski, R. Gadde, S. Zagoruyko, G. Obozinski, and R. Marlet, "A MRF Shape Prior for Facade Parsing With Occlusions," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[113] B. Fröhlich, E. Rodner, M. Kemmler, and J. Denzler, "Large-scale Gaussian Process Multi-class Classification for Semantic Segmentation and Facade Recognition," *Mach. Vision Appl.*, vol. 24, pp. 1043–1053, July 2013.

[114] K. van de Sande, T. Gevers, and C. Snoek, "Evaluating Color Descriptors for Object and Scene Recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, pp. 1582–1596, Sept. 2010.

[115] K. Terzić and B. Neumann, "Integrating Context Priors into a Decision Tree Classification Scheme," in *International Conference on Machine Vision, Image Processing, and Pattern Analysis*, (Bangkok, Thailand), Dec. 2009.

[116] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Monterey, CA: Wadsworth and Brooks, 1984.

[117] M. Drauschke and W. Förstner, "Comparison of Adaboost and ADTboost for Feature Subset Selection," in *Pattern Recognition in Information Systems, Proceedings of the 8th International Workshop on Pattern Recognition in Information Systems, {PRIS} 2008, In conjunction with {ICEIS} 2008, Barcelona, Spain, June 2008* (A. Juan-Císcar and G. Sánchez-Albaladejo, eds.), pp. 113–122, {INSTICC} {PRESS}, 2008.

[118] M. Y. Yang, W. Förstner, and D. Chai, "Feature Evaluation for Building Facade Images – An Empirical Study," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XXXIX-B3, pp. 513–518, 2012.

[119] M. Y. Yang and W. Förstner, "Regionwise Classification of Building Facade Images," in *Photogrammetric Image Analysis - {ISPRS} Conference, {PIA} 2011, Munich, Germany, October 5-7, 2011. Proceedings* (U. Stilla, F. Rottensteiner, H. Mayer, B. Jutzi, and M. Butenuth, eds.), vol. 6952 of *Lecture Notes in Computer Science*, pp. 209–220, Springer, 2011.

[120] V. Jampani, R. Gadde, and P. V. Gehler, "Efficient 2D and 3D Facade Segmentation Using Auto-context," in *Proceedings of the 2015 IEEE Winter Conference on Applications of Computer Vision*, WACV '15, (Washington, DC, USA), pp. 1038–1045, IEEE Computer Society, 2015.

[121] V. Kolmogorov and R. Zabih, "What Energy Functions Can Be Minimized via Graph Cuts?," in *Proceedings of the 7th European Conference on Computer Vision-Part III*, ECCV '02, (London, UK, UK), pp. 65–81, Springer-Verlag, 2002.

[122] L. Gorelick, Y. Boykov, O. Veksler, I. B. Ayed, and A. Delong, "Submodularization for Binary Pairwise Energies.," in *CVPR*, pp. 1154–1161, IEEE Computer Society, 2014.

[123] Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization via Graph Cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, pp. 1222–1239, Nov. 2001.

[124] A. Delong, A. Osokin, H. N. Isack, and Y. Boykov, "Fast Approximate Energy Minimization with Label Costs," *Int. J. Comput. Vision*, vol. 96, no. 1, pp. 1–27, 2012.

[125] B. Fröhlich, E. Rodner, and J. Denzler, "A Fast Approach for Pixelwise Labeling of Facade Images," in *20th International Conference on Pattern Recognition, {ICPR} 2010, Istanbul, Turkey, 23-26 August 2010*, pp. 3029–3032, {IEEE} Computer Society, 2010.

[126] Andelo Martinovic and Luc Van Gool, "Bayesian Grammar Learning for Inverse Procedural Modeling," *2013 IEEE Conference on Computer Vision and Pattern Recognition*, vol. 00, pp. 201–208, 2013.

[127] R. Socher, C. C. Lin, A. Y. Ng, and C. D. Manning, "Parsing Natural Scenes and Natural Language with Recursive Neural Networks," in *Proceedings of the 26th International Conference on Machine Learning (ICML)*, 2011.

[128] P. Dollar, Z. Tu, P. Perona, and S. Belongie, "Integral Channel Features," in *Proc. BMVC*, pp. 91.1–91.11, 2009.

[129] O. Teboul, L. Simon, P. Koutsourakis, and N. Paragios, "Segmentation of building facades using procedural shape priors.," in *CVPR*, pp. 3105–3112, IEEE Computer Society, 2010.

[130] H. Riemenschneider, U. Krispel, W. Thaller, M. Donoser, S. Havemann, D. W. Fellner, and H. Bischof, "Irregular lattices for complex shape grammar facade parsing.," in *CVPR*, pp. 1640–1647, IEEE Computer Society, 2012.

[131] M. Mathias, A. Martinović, J. Weissenberg, and L. Van Gool, "Procedural 3D Building Reconstruction Using Shape Grammars and Detectors," in *3DIMPVT*, pp. 304–311, 2011.

[132] R. Gadde, R. Marlet, and N. Paragios, "Learning Grammars for Architecture-Specific Facade Parsing," *International Journal of Computer Vision*, vol. 117, no. 3, pp. 290–316, 2016.

[133] G. Valiente, *Algorithms on trees and graphs*. Berlin; New York: Springer, 2002.

[134] N. Komodakis, N. Paragios, and G. Tziritas, "Clustering via LP-based Stabilities," in *Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 8-11, 2008*, pp. 865–872, 2008.

[135] D. L. Davies and D. W. Bouldin, "A Cluster Separation Measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 1, pp. 224–227, Feb. 1979.

[136] C. Legány, S. Juhász, and A. Babos, "Cluster validity measurement techniques," in *Proceedings of the 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases*, AIKED'06, (Stevens Point, Wisconsin, USA), pp. 388–393, World Scientific and Engineering Academy and Society (WSEAS), 2006.

[137] J. Weissenberg, H. Riemenschneider, M. Prasad, and L. Van Gool, "Is There a Procedural Logic to Architecture?," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '13, (Washington, DC, USA), pp. 185–192, IEEE Computer Society, 2013.

[138] M. Kozinski and R. Marlet, "Image parsing with graph grammars and Markov Random Fields applied to facade analysis," in {*IEEE*} *Winter Conference on Applications of Computer Vision, Steamboat Springs, CO, USA, March 24-26, 2014*, pp. 729–736, 2014.

[139] S. T. Teoh, "Generalized Descriptions for the Procedural Modeling of Ancient East Asian Buildings," in *Proceedings of the Fifth Eurographics Conference on Computational Aesthetics in Graphics, Visualization and Imaging*, Computational Aesthetics'09, (Aire-la-Ville, Switzerland, Switzerland), pp. 17–24, Eurographics Association, 2009.

[140] M. Park, K. Brocklehurst, R. T. Collins, and Y. Liu, "Translation-symmetry-based Perceptual Grouping with Applications to Urban Scenes," in *Proceedings of the 10th Asian Conference on Computer Vision - Volume Part III*, ACCV'10, (Berlin, Heidelberg), pp. 329–342, Springer-Verlag, 2011.

[141] S. Wenzel, M. Drauschke, and W. Förstner, "Detection of repeated structures in facade images," *Pattern Recognition and Image Analysis*, vol. 18, no. 3, pp. 406–411, 2008.

[142] G. Loy and J.-O. Eklundh, "Detecting Symmetry and Symmetric Constellations of Features.," in *ECCV (2)* (A. Leonardis, H. Bischof, and A. Pinz, eds.), vol. 3952 of *Lecture Notes in Computer Science*, pp. 508–521, Springer, 2006.

[143] M. Pollefeys, "Discovering and Exploiting 3D Symmetries in Structure from Motion," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, (Washington, DC, USA), pp. 1514–1521, IEEE Computer Society, 2012.

[144] M. Kushnir and I. Shimshoni, "Epipolar Geometry Estimation for Urban Scenes with Repetitive Structures," in *Proceedings of the 11th Asian Conference on Computer Vision - Volume Part IV*, ACCV'12, (Berlin, Heidelberg), pp. 163–176, Springer-Verlag, 2013.

[145] L. Nan, A. Sharf, H. Zhang, D. Cohen-Or, and B. Chen, "SmartBoxes for Interactive Urban Reconstruction," in *ACM SIGGRAPH 2010 Papers*, SIGGRAPH '10, (New York, NY, USA), pp. 93:1—-93:10, ACM, 2010.

[146] S. Friedman and I. Stamos, "Online Detection of Repeated Structures in Point Clouds of Urban Scenes for Compression and Registration," *Int. J. Comput. Vision*, vol. 102, pp. 112–128, Mar. 2013.

[147] C. Wu, J.-M. Frahm, and M. Pollefeys, "Detecting Large Repetitive Structures with Salient Boundaries," in *Proceedings of the 11th European Conference on Computer Vision: Part II*, ECCV'10, (Berlin, Heidelberg), pp. 142–155, Springer-Verlag, 2010.

[148] J. Wang, C. Liu, T. Shen, and L. Quan, "Structure-driven facade parsing with irregular patterns," in *3rd {IAPR} Asian Conference on Pattern Recognition, {ACPR} 2015, Kuala Lumpur, Malaysia, November 3-6, 2015*, pp. 41–45, 2015.

[149] P. J. Green, "Reversible jump {Markov} chain {Monte Carlo} computation and {Bayesian} model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.

[150] T. Han, C. Liu, C.-L. Tai, and L. Quan, "Quasi-regular Facade Structure Extraction," in *Computer Vision - {ACCV} 2012, 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part {IV}* (K. M. Lee, Y. Matsushita, J. M. Rehg, and Z. Hu, eds.), vol. 7727 of *Lecture Notes in Computer Science*, pp. 552–564, Springer, 2012.

[151] D. Dai, H. Riemenschneider, G. Schmitt, and L. Van, "Example-Based Facade Texture Synthesis," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*, ICCV '13, (Washington, DC, USA), pp. 1065–1072, IEEE Computer Society, 2013.

[152] H. F. Xiao, G. F. Meng, L. F. Wang, S. M. Xiang, and C. H. Pan, "Facade repetition extraction using block matrix based model," pp. 1673–1677, 2014.

[153] P. Zhao, L. Yang, H. Zhang, and L. Quan, "Per-pixel translational symmetry detection, optimization, and segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, June 2012.

[154] C.-L. Tai, "Parsing FaçAde with Rank-one Approximation," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, (Washington, DC, USA), pp. 1720–1727, IEEE Computer Society, 2012.

[155] S.-C. Zhu, C.-E. Guo, Y. Wang, and Z. Xu, "What Are Textons?," *Int. J. Comput. Vision*, vol. 62, pp. 121–143, Apr. 2005.

[156] A. Cohen, A. G. Schwing, and M. Pollefeys, "Efficient Structured Parsing of Facades Using Dynamic Programming," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[157] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr, "Associative Hierarchical Random Fields," {*IEEE*} *Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 6, pp. 1056–1077, 2014.

[158] L. R. Ford and D. R. Fulkerson, *Maximal Flow Through a Network*, pp. 243–248. Boston, MA: Birkh{ä}user Boston, 1987.

[159] C. Rother, V. Kolmogorov, and A. Blake, ""GrabCut": Interactive Foreground Extraction Using Iterated Graph Cuts," *ACM Trans. Graph.*, vol. 23, pp. 309–314, Aug. 2004.

[160] C. Göring, B. Fröhlich, and J. Denzler, "Semantic Segmentation using GrabCut.," in *VISAPP (1)*, pp. 597–602, 2012.

[161] N. B. S. Vu, *Image segmentation with semantic priors: A graph cut approach*. PhD thesis, UNIVERSITY OF CALIFORNIA Santa Barbara, 2008.

[162] B. Peng and O. Veksler, "Parameter Selection for Graph Cut Based Image Segmentation," in *Proc. BMVC*, pp. 16.1–16.10, 2008.

[163] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *J. Comput. Syst. Sci.*, vol. 55, pp. 119–139, Aug. 1997.

[164] X. Ren and J. Malik, "Learning a Classification Model for Segmentation," in *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV '03, (Washington, DC, USA), pp. 10—-, IEEE Computer Society, 2003.

[165] K. O. Oyebode and J. R. Tapamo, "ADAPTIVE PARAMETER SELECTION FOR GRAPH CUT-BASED SEGMENTATION ON CELL IMAGES," *Image Analysis & Stereology*, vol. 35, no. 1, pp. 29–37, 2016.

[166] S. Candemir, K. Palaniappan, and Y. S. Akgul, "Multi-class regularization parameter learning for graph cut image segmentation," in *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on*, pp. 1473–1476, IEEE, 2013.

[167] J. Friedman, T. Hastie, and R. Tibshirani, "Additive Logistic Regression: a Statistical View of Boosting," *The Annals of Statistics*, vol. 38, no. 2, 2000.

[168] C. Rother, V. Kolmogorov, V. S. Lempitsky, and M. Szummer, "Optimizing Binary MRFs via Extended Roof Duality.," in *CVPR*, IEEE Computer Society, 2007.

[169] B. Savchynskyy, J. H. Kappes, P. Swoboda, and C. Schnörr, "Global MAP-Optimality by Shrinking the Combinatorial Search Area with Convex Relaxation," in *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pp. 1950–1958, 2013.

[170] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael, "Learning Low-Level Vision," *International Journal of Computer Vision*, vol. 40, no. 1, pp. 25–47, 2000.

[171] K. Duan, D. Parikh, D. J. Crandall, and K. Grauman, "Discovering localized attributes for fine-grained recognition.," in *CVPR*, pp. 3474–3481, IEEE Computer Society, 2012.

[172] J. H. Kappes, B. Andres, F. A. Hamprecht, C. Schnörr, S. Nowozin, D. Batra, S. Kim, B. X. Kausler, T. Kröger, J. Lellmann, N. Komodakis, B. Savchynskyy, and C. Rother, "A Com-

parative Study of Modern Inference Techniques for Structured Discrete Energy Minimization Problems.," *International Journal of Computer Vision*, vol. 115, no. 2, pp. 155–184, 2015.

[173] N. D. Campbell, G. Vogiatzis, C. Hernández, and R. Cipolla, "Using Multiple Hypotheses to Improve Depth-Maps for Multi-View Stereo," in *Proceedings of the 10th European Conference on Computer Vision: Part I*, ECCV '08, (Berlin, Heidelberg), pp. 766–779, Springer-Verlag, 2008.

[174] G. E. Hinton, "Training Products of Experts by Minimizing Contrastive Divergence," *Neural Comput.*, vol. 14, pp. 1771–1800, Aug. 2002.

[175] S. M. A. Eslami, N. Heess, C. K. I. Williams, and J. Winn, "The Shape Boltzmann Machine: a Strong Model of Object Shape," in *International Journal of Computer Vision*, 2013.

[176] R. Salakhutdinov and G. Hinton, "Deep Boltzmann Machines," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, vol. 5, pp. 448–455, 2009.

[177] I. Goodfellow, M. Mirza, A. Courville, and Y. Bengio, "Multi-Prediction Deep Boltzmann Machines," in *Advances in Neural Information Processing Systems 26* (C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds.), pp. 548–556, Curran Associates, Inc., 2013.

[178] S. Eslami and C. Williams, "A Generative Model for Parts-based Object Segmentation," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 100–107, Curran Associates, Inc., 2012.

[179] M. E. A. V. D. Kirillov Alexander;Gavrikov, "Deep Part-Based Generative Shape Model with Latent Variables," in *British Machine Vision Conference*, 2016.

[180] A. Kae, K. Sohn, H. Lee, and E. Learned-Miller, "Augmenting CRFs with Boltzmann Machine Shape Priors for Image Labeling," University of Massachusetts Amherst and University of Michigan Ann Arbor, 2013.

[181] E. E. F. M. T. T. M. Cetin, "Disjunctive Normal Shape Boltzmann Machine," 2017.

[182] N. Heess, N. L. Roux, and J. M. Winn, "Weakly Supervised Learning of Foreground-Background Segmentation using Masked RBMs," *CoRR*, vol. abs/1107.3823, 2011.

[183] N. L. Roux, N. Heess, J. Shotton, and J. M. Winn, "Learning a Generative Model of Images by Factoring Appearance and Shape.," *Neural Computation*, vol. 23, no. 3, pp. 593–650, 2011.

[184] T. Tieleman, "Training Restricted Boltzmann Machines Using Approximations to the Likelihood Gradient," in *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, (New York, NY, USA), pp. 1064–1071, ACM, 2008.

[185] D. Turcsany, A. Bargiela, and T. Maul, "Local Receptive Field Constrained Deep Networks," *Inf. Sci.*, vol. 349, pp. 229–247, July 2016.

[186] B. Taskar, C. Guestrin, and D. Koller, "Max-margin Markov Networks," in *Proceedings of the 16th International Conference on Neural Information Processing Systems*, NIPS'03, (Cambridge, MA, USA), pp. 25–32, MIT Press, 2003.

[187] J. Yang, S. Sáfár, and M. H. Yang, "Max-Margin Boltzmann Machines for Object Segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 320–327, June 2014.

[188] C.-N. J. Yu and T. Joachims, "Learning Structural SVMs with Latent Variables," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, (New York, NY, USA), pp. 1169–1176, ACM, 2009.

[189] Y. Kihara, M. Soloviev, and T. Chen, "In the Shadows, Shape Priors Shine: Using Occlusion to Improve Multi-Region Segmentation," *CoRR*, vol. abs/1606.04590, 2016.

[190] M. Nitzberg and D. Mumford, "The 2.1-D sketch," in *Third International Conference on Computer Vision, {ICCV} 1990. Osaka, Japan, 4-7 December, 1990, Proceedings*, pp. 138–144, IEEE, 1990.

[191] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 2528–2535, IEEE, 2010.

[192] V. Badrinarayanan, A. Handa, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling.," *CoRR*, vol. abs/1505.07293, 2015.

[193] D. Wu, L. Pigou, P.-J. Kindermans, N. D.-H. Le, L. Shao, J. Dambre, and J.-M. Odobez, "Deep dynamic neural networks for multimodal gesture segmentation and recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 8, pp. 1583–1597, 2016.

[194] S. Li, "Markov random field models in computer vision," in *Third European Conference on Computer Vision , Part II.*, pp. 361–370, 1994.

[195] A. k. Björck, *Numerical Methods for Least Squares Problems*. Siam Philadelphia, 1996.

[196] P. H. S. Torr and D. W. Murray, "The Development and Comparison of Robust Methodsfor Estimating the Fundamental Matrix," *Int. J. Comput. Vision*, vol. 24, pp. 271–300, Sept. 1997.

[197] H. Isack and Y. Boykov, "Energy-Based Geometric Multi-model Fitting," *Int. J. Comput. Vision*, vol. 97, pp. 123–147, Apr. 2012.

[198] C. Stewart, "Bias in Robust Estimation caused by discontinuities and multiple structures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 8, pp. 331–338, 1997.

[199] P. H. S. Torr and A. Zisserman, "MLESAC : A new robust estimator with application to estimating image geometry," *Computer Vision and Image Understanding - Special issue on robusst statistical techniques in image understanding*, vol. 78, no. 1, pp. 138–156, 2000.

[200] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm, "Usac: A universal framework for random sample consensus.," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 2022–2038, 2013.

[201] S. Li, "A 1.488 approximation algorithm for the uncapacitated facility location problem," in *Proceedings of the 38th international conference on Automata, languages and programming*, ICALP'11, (Berlin, Heidelberg), pp. 77–88, Springer-Verlag, 2011.

[202] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. on Automatic Control*, vol. 19, no. 6, pp. 716–723, 1974.

[203] T. T. Pham, T.-j. Chin, J. Yu, and D. Suter, "Simultaneous Sampling and Multi-Structure Fitting with Adaptive Reversible Jump MCMC," in *Advances in Neural Information Processing Systems 24*, pp. 540–548, 2011.

[204] N. Thakoor and J. Gao, "Branch-and-bound hypothesis selection for two-view multiple structure and motion segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition, (CVPR).*, pp. 1–6, IEEE, 2008.

[205] H. Li, "Two-view motion segmentation from linear programming relaxation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*, pp. 1–8, IEEE, 2007.

[206] P. H. S. Torr, "Geometric motion segmentation and model selection," *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 356, no. 1740, pp. 1321–1340, 1998.

[207] J. Yu, T.-J. Chin, and D. Suter, "A global optimization approach to robust multi-model fitting," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2041–2048, IEEE, 2011.

[208] R. Toldo and A. Fusiello, "Robust Multiple Structures Estimation with J-Linkage," in *Proceedings of the 10th European Conference on Computer Vision: Part I*, ECCV '08, (Berlin, Heidelberg), pp. 537–547, Springer-Verlag, 2008.

[209] L. Xu, E. Oja, and P. Kultanen, "A new curve detection method: Randomized Hough transform (RHT)," *Pattern Recognition Letters*, vol. 11, no. 5, pp. 331–338, 1990.

[210] W. Zhang and J. Kosecka, "Nonparametric estimation of multiple structures with outliers," in *Proceedings of the international conference on Dynamical vision*, pp. 60–74, 2006.

[211] T.-J. Chin, J. Yu, and D. Suter, "Accelerated Hypothesis Generation for Multistructure Data via Preference Analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, pp. 625–638, Apr. 2012.

[212] Y. Kanazawa and H. Kawakami, "Detection of planar regions with uncalibrated stereo using distributions of feature points," in *British Machine Vision Conference*, vol. 1, pp. 247–256, Citeseer, 2004.

[213] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd ed., 2009.

[214] R. Hoseinnezhad, B.-N. Vo, and D. Suter, "Fast segmentation of multiple motions," *Cognitive systems*, 2009.

[215] M. Tepper, P. Musé, and A. Almansa, "Meaningful Clustered Forest: an Automatic and Robust Clustering Algorithm," *CoRR*, vol. abs/1104.0, 2011.

[216] F. R. B. M. I. Jordan, "Learning spectral clustering," *Advances in Neural Information Processing Systems*, vol. 16, p. 305, 2004.

[217] B. Mohar and Y. Alavi, "The Laplacian spectrum of graphs," *Graph theory, combinatorics, and applications*, vol. 2, pp. 871–898, 1991.

[218] F. R. K. Chung, "Spectral Graph Theory," *CBMS Regional Conference Series in Mathematics*, vol. 92, 1997.

[219] E. Dimitriadou and S. Dolnicar, "An Examination of Indexes for Determining the Number of Clusters in Binary Data Sets," *Psychometrika*, vol. 67, no. 1, pp. 137–160, 2002.

[220] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[221] R. Gadde, V. Jampani, R. Marlet, and P. V. Gehler, "Efficient 2D and 3D Facade Segmentation using Auto-Context," *CoRR*, vol. abs/1606.0, 2016.

[222] C. A. Coello Coello and M. S. Lechuga, "MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization," in *Proceedings of the Evolutionary Computation on 2002. CEC '02. Proceedings of the 2002 Congress - Volume 02*, CEC '02, (Washington, DC, USA), pp. 1051–1056, IEEE Computer Society, 2002.

[223] A. Banks, J. Vincent, and C. Anyakoha, "A review of particle swarm optimization. Part II: hybridisation, combinatorial, multicriteria and constrained optimization, and indicative applications," *Natural Computing*, vol. 7, no. 1, pp. 109–124, 2008.

[224] O. Teboul, I. Kokkinos, L. Simon, P. Koutsourakis, and N. Paragios, "Shape grammar parsing via Reinforcement Learning.," in *CVPR*, pp. 2273–2280, IEEE Computer Society, 2011.

[225] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," in *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, 1998.

[226] A. Torralba, "Contextual Priming for Object Detection," *International Journal of Computer Vision*, vol. 53, no. 2, pp. 169–191, 2003.

[227] F. Tabib Mahmoudi, F. Samadzadegan, and P. Reinartz, "Context Aware Modification on the Object Based Image Analysis," *Journal of the Indian Society of Remote Sensing*, vol. 43, no. 4, pp. 709–717, 2015.

[228] Y. Tang, "Gated Boltzmann Machine for Recognition under Occlusion," in *NIPS Workshop on Transfer Learning by Learning Rich Generative Models*, 2010.

[229] S. Tsogkas, I. Kokkinos, G. Papandreou, and A. Vedaldi, "Semantic Part Segmentation with Deep Learning," *CoRR*, vol. abs/1505.0, 2015.

[230] S. Gutstein, O. Fuentes, and E. Freudenthal, "Knowledge Transfer in Deep convolutional Neural Nets.," *International Journal on Artificial Intelligence Tools*, vol. 17, no. 3, pp. 555–567, 2008.

[231] G. E. Hinton, "To Recognize Shapes First Learn to Generate Images," in *Computational Neuroscience: Theoretical Insights into Brain Function*, Elsevier, 2007.

[232] G. E. Hinton, *A Practical Guide to Training Restricted Boltzmann Machines*, pp. 599–619. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[233] M. A. Keyvanrad and M. M. Homayounpour, "Deep Belief Network Training Improvement Using Elite Samples Minimizing Free Energy," *CoRR*, vol. abs/1411.4, 2014.

[234] D. H. Wolpert, "Stacked generalization," *Neural networks*, vol. 5, no. 2, pp. 241–259, 1992.

[235] C. Huang, Y. Li, C. C. Loy, and X. Tang, "Learning Deep Representation for Imbalanced Classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[236] S. J. Raudys and A. K. Jain, "Small sample size effects in statistical pattern recognition: recommendations for practitioners," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, pp. 252–264, Mar. 1991.

[237] M. F. Mø ller, "A scaled conjugate gradient algorithm for fast supervised learning," *NEURAL NETWORKS*, vol. 6, no. 4, pp. 525–533, 1993.

[238] H. Peng, B. Roysam, and G. A. Ascoli, "Automated image computing reshapes computational neuroscience," {*BMC*} *Bioinformatics*, vol. 14, p. 293, 2013.

[239] D. Robben, E. Türetken, S. Sunaert, V. Thijs, G. Wilms, P. Fua, F. Maes, and P. Suetens, *Simultaneous Segmentation and Anatomical Labeling of the Cerebral Vasculature*, pp. 307–314. Cham: Springer International Publishing, 2014.

[240] N. M. O. Heess, "Learning generative models of mid-level structure in natural images," 2012.

[241] A. Ioannidou, E. Chatzilari, S. Nikolopoulos, and I. Kompatsiaris, "Deep learning advances in computer vision with 3d data: A survey," *ACM Comput. Surv.*, vol. 50, pp. 20:1–20:38, Apr. 2017.