

---

# End-to-end example-based sim-to-real RL policy transfer based on neural stylisation with application to robotic cutting

---

Received: 9 May 2025

Accepted: 23 February 2026

Published online: 12 March 2026

Cite this article as: Hathaway J., Rastegarpanah A. & Stolkin R. End-to-end example-based sim-to-real RL policy transfer based on neural stylisation with application to robotic cutting. *Sci Rep* (2026). <https://doi.org/10.1038/s41598-026-41735-5>

Jamie Hathaway, Alireza Rastegarpanah & Rustam Stolkin

We are providing an unedited version of this manuscript to give early access to its findings. Before final publication, the manuscript will undergo further editing. Please note there may be errors present which affect the content, and all legal disclaimers apply.

If this paper is publishing under a Transparent Peer Review model then Peer Review reports will publish with the final article.

ARTICLE IN PRESS

# End-to-end example-based sim-to-real RL policy transfer based on neural stylisation with application to robotic cutting

Jamie Hathaway<sup>1,2,†</sup>, Alireza Rastegarpanah<sup>1,3,†,\*</sup>, and Rustam Stolkin<sup>1,2</sup>

<sup>1</sup>Extreme Robotics Lab, School of Metallurgy and Materials, University of Birmingham, Birmingham, B15 2TT, UK

<sup>2</sup>The Faraday Institution, Quad One, Harwell Science and Innovation Campus, Didcot, OX11 0RA, UK

<sup>3</sup>School of Computer Science and Digital Technologies, Aston University, Birmingham B4 7ET, UK

\*Corresponding Author: Alireza Rastegarpanah, Email: a.rastegarpanah@bham.ac.uk

†Jamie Hathaway and Alireza Rastegarpanah are both recognised as joint first authors.

## ABSTRACT

Whereas reinforcement learning has been applied with success to a range of robotic control problems in complex, uncertain environments, reliance on extensive data - typically sourced from simulation environments - limits real-world deployment due to the domain gap between simulated and physical systems, coupled with limited real-world sample availability. We propose a novel method for sim-to-real transfer of reinforcement learning policies, based on a reinterpretation of neural style transfer from image processing to synthesise novel training data from unpaired unlabelled real world datasets. We employ a variational autoencoder to jointly learn self-supervised feature representations for style transfer and generate weakly paired source-target trajectories to improve physical realism of synthesised trajectories. We demonstrate the application of our approach based on the case study of robot cutting of unknown materials. Compared to baseline methods, including our previous work, CycleGAN, and conditional variational autoencoder-based time series translation, our approach achieves improved task completion time and behavioural stability with minimal real-world data. Our framework demonstrates robustness to geometric and material variation, and highlights the feasibility of policy adaptation in challenging contact-rich tasks where real-world reward information is unavailable.

## Introduction

Emerging applications for robotics have fostered increasing interest in low-volume, high-mix disassembly processes in industry. These processes are characterised by a high degree of uncertainty, due to variable designs and lack of prior knowledge in the form of component models, coupled with demands of logistical flexibility, which traditionally implies the requirement for expensive reprogramming and reconfiguration of robots. This is of interest in domains such as nuclear decommissioning, robotic disassembly of complex products for recycling and re-use, and even areas such as robotic surgery or demolition with robotised demolition equipment. Nonetheless, challenges exist in automated planning and task execution for destructive operations. Whereas manufacturing paradigms centre around achieving high dimensional tolerances and precise control on a known workpiece, for disassembly, the precise location of cutting is less important (few mm as opposed to  $\mu\text{m}$ ) while the precise sequence of cutting operations may not be known in advance. This uncertainty has motivated various approaches to robotic cutting, consisting of goal-conditioned trial-and-error & revision<sup>1,2</sup>, 3D reconstruction & planning<sup>3</sup>, and online learning & adaptation<sup>4,5</sup>.

Reinforcement learning (RL) has been applied with success to a variety of contact-rich tasks<sup>6,7</sup>, including robotic cutting<sup>8,9</sup>, particularly with difficult-to-model environments with complex robot-environment interactions, but are nonetheless data intensive. Whereas simulation environments offer reduced complexity and overhead of data collection, differences between simulated and physical cutting processes limit the applicability of adaptive methods to real-world tasks. Examples of such differences include motor backlash, tool wear, chattering, cross-domain mismatch of process and model parameters and other disturbances. These differences motivate the use of domain adaptation methods to align representations or behaviours across domains with minimal real-world supervision. These can be broadly separated into unified feature representation learning, model-based correction, and model-free synthesis of target domain examples.

Domain adaptive methods include<sup>10</sup> in which policies are trained on a cross-domain latent feature representation by aligning source and target domain distributions. A related concept applied to milling was proposed in<sup>8</sup> based on a cross-domain meta-model, trained on pairwise unified feature representations. Similarly, adversarial losses using domain discriminators have been employed for cross-domain tool wear classification<sup>11</sup>. Reconstruction-based methods have also been employed to

jointly model observation and class distributions<sup>12</sup>; this concept has been further developed based on conditional variational autoencoders (CVAEs)<sup>13</sup> wherein CVAE feature representations were used to train an RL policy, while feature representations are aligned across domains.

Model-based approaches have previously also been employed for domain adaptation, wherein a source domain task model is augmented with a corrective model based on physics-informed approaches<sup>14</sup>, neural networks<sup>15,16</sup> or Gaussian process (GP) models<sup>17,18</sup> learned from target domain data. In our previous work,<sup>19</sup> we proposed an imitation learning framework in which a GP corrective model was learned from multiple cutting trajectories. Nonetheless, model-based approaches incur limitations of modelling assumptions under which the models are introduced, and incur a dataset overhead, particularly for deep predictive modelling approaches.

Relating to the aforementioned approaches is direct alignment of observations across domains via translation or generative models. In the context of milling,<sup>20</sup> proposed a domain adaptation method for condition monitoring of different milling tools based on a generative CNN. Similarly,<sup>21</sup> proposed a domain adaptive imitation learning framework from visual demonstrations based on CycleGAN<sup>22</sup>. Generation at object level has also been proposed<sup>23</sup> wherein a StyleGAN image translation model is trained object-wise on weakly-paired cross-domain datasets for 6D pose estimation. CVAEs have also been employed for domain adaptation via synthesis of novel target domain examples<sup>24</sup>.

Neural style transfer has been extensively researched in the context of image processing<sup>25,26</sup>. Recently, this concept has been extended to motion execution. Thus far, its application has been limited largely to expressive stylised motions mirroring that of human operators<sup>27,28</sup>. Nonetheless, its applicability to synthesise novel trajectories with characteristics of diverse human operators presents a compelling case for its application to other domain adaptation problems. Recently, this has been applied for dataset augmentation tasks<sup>29</sup>. A limitation of the aforementioned methods is lack of a suitable pairing mechanism for style and content, as well as lack of feature extractor backbones prevalent in image processing tasks. For transfer learning, addressed this problem<sup>30</sup> by building on the concept of conditional adversarial domain adaptation<sup>31</sup> to achieve feature-level style transfer for transfer learning. Nonetheless, adversarial alignment can be difficult to train, with well-known problems of mode collapse and vanishing gradients. Whereas these developments have been applied to time series classification problems, application of style transfer for RL policy transfer is, to the best of our knowledge, largely unexplored.

This paper extends our previous example-based approach for sim-to-real adaptation to arbitrary real world examples. As with our previous work, our approach does not require re-training of classifiers or encoder networks to adapt to new scenarios (different disturbance forces, differing sensor dynamics, etc.). In contrast to prior work that applies neural style transfer primarily for stylised motion synthesis or dataset augmentation, we apply it as a trajectory-level domain adaptation mechanism for robotic skill transfer. Our contributions are threefold: (1) a latent-space pairing mechanism for content and style that operates without paired examples or retraining; (2) a novel transfer framework based on neural style transfer that does not require labelled or reward-supervised data from the target domain; and (3) empirical evaluation on robotic cutting, a task where conventional reinforcement learning pipelines are difficult to apply due to the absence of reward signal in the real-world deployment environment. An overview of our framework is provided in Figure 1.

## Style transfer framework

### Variational autoencoder

The variational autoencoder (VAE) consists of two neural networks: an encoder  $q_\phi(z|x)$  that approximates the posterior over latent variables  $z \in \mathbb{R}^L$ , and a decoder  $p_\theta(x|z)$  that reconstructs the data from the latent representation. The encoder network  $q_\phi(z|x)$  outputs distributional parameters  $\mu_\phi(x)$  and diagonal log-variance  $\log \sigma_\phi^2(x)$  of a multivariate Gaussian posterior as

$$q_\phi(z|x) = \mathcal{N}(z; \mu_\phi(x), \text{diag}(\sigma_\phi(x))) \quad (1)$$

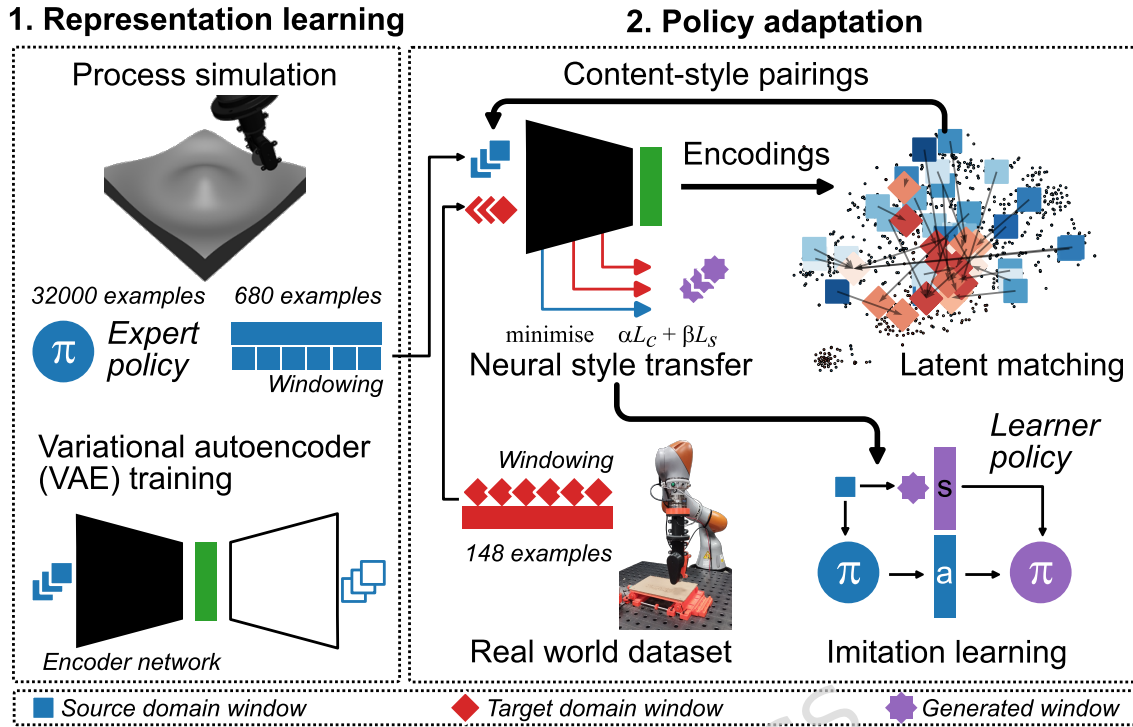
A latent code  $z$  is sampled via the reparametrisation trick:

$$z = \mu_\phi(x) + \sigma_\phi(x) \odot \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, I). \quad (2)$$

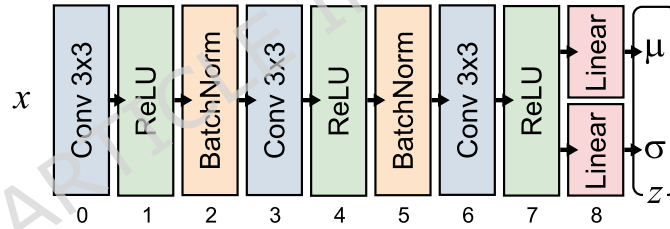
The decoder  $p_\theta(x|z)$  reconstructs the input  $x$  from the latent code; for continuous data, we used an isotropic Gaussian likelihood  $\mathcal{N}(x; \mu_\theta(z), I)$ . The VAE loss function is expressed as the evidence lower bound (ELBO), which comprises a reconstruction loss and a KL divergence regularising term:

$$\mathcal{L}(\theta, \phi; x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - D_{\text{KL}}[q_\phi(z|x) \| p(z)] \quad (3)$$

where  $p(z) = \mathcal{N}(0, I)$  is the standard normal prior over latent codes. Training was carried out with the Adam optimiser, with model hyperparameters established via manual search, reported in Table 1. Both encoder and decoder networks were



**Figure 1.** Overview of proposed framework. In the first stage, a simulation of cutting mechanics is used to generate an expert policy and a variational autoencoder (VAE) is trained on simulated trajectory windows. In the second stage, the VAE encoded representations are used to generate pairings between a simulated and real world dataset which are used as style targets. Finally, expert trajectories are used to train a learner target domain policy with the generated observation windows.



**Figure 2.** Overview of VAE encoder architecture; layer indices for style transfer are demarcated.

implemented as strided convolutional networks with 3 layers. Batch normalisation was further employed to accelerate convergence and reduce training instability. The encoder architecture is visualised in Figure 2.

The VAE training dataset consisted of a mixture of 680 on-policy and off-policy simulated trajectories. We consider a trajectory as a multivariate time series of length  $T$  which comprises a sequence of state-action pairs:

$$\tau_j = \{(x_t, y_t)\}_{t=1}^T \quad (4)$$

where  $x_t \in \mathbb{R}^{N_s}$ ,  $y_t \in \mathbb{R}^{N_a}$  are the states, actions at time  $t$  respectively. The index  $j \in \{1 \dots N_\tau\}$  denotes the trajectory, with  $N_\tau$  the total number of trajectories in the dataset. Each trajectory  $\tau_j$  is divided into overlapping windows of length  $N$ , resulting in a set of state and action windows:

$$x^{(i)} = [x_t, x_{t+1}, \dots, x_{t+N-1}] \in \mathbb{R}^{N \times N_s} \quad (5)$$

$$y^{(i)} = [y_t, y_{t+1}, \dots, y_{t+N-1}] \in \mathbb{R}^{N \times N_a} \quad (6)$$

where the  $i$ th window corresponds to  $i = j + t$ , and  $t \in \{1 \dots T - N + 1\}$  to avoid windows overrunning the trajectory. The window width  $N$  and latent code dimensionality emerge as tunable parameters for which a trade-off exists between the temporal context afforded to the model, reproduction accuracy and saliency of the latent space. Through preliminary experiments,

this was reflected in increased RMS error of the autoencoder reconstructions and reduced average cosine similarity between simulated and real world embeddings with increasing  $N$  and dimensionality respectively. A window size of  $N = 100$  samples (2 seconds) was identified as providing the best trade-off between these factors.

### Policy adaptation

We adopt a similar approach to our previous work<sup>19</sup> to adapt a pre-trained policy to observations synthesised from unlabelled target domain data. In this procedure, an ‘‘expert’’ policy  $\pi_e$  is initially trained in a simulation environment with a physically-informed cutting model, as introduced in our previous work<sup>32</sup>, with model parameters from Table 2. The expert was trained initially for 32000 episodes using the proximal policy optimisation (PPO) algorithm with domain randomisation of material properties. The workstation for training consisted of an Intel i7-8086K CPU, NVIDIA GTX 1080 Ti GPU with 11GB VRAM, and 32GB RAM. To synthesise a training dataset to adapt the policy to the target domain, a translation function  $f : \mathbb{R}^{N \times N_S} \rightarrow \mathbb{R}^{N \times N_A}$  is applied to each simulated state window:

$$x^{(g,i)*} = f(x^{(c,i)}) \quad (7)$$

where we refer to simulated windows as  $c$ , real world windows as  $s$ , and synthesised windows as  $g$ . We refer to such synthesised observations  $g$  as the ‘‘surrogate target domain’’. Synthesised states are paired with the corresponding expert action on  $x^{(c)}$  to generate a labelled dataset

$$\mathcal{D} = \{(x^{(g,i)}, \pi_e(x^{(c,i)}))\}. \quad (8)$$

We subsequently train a target domain policy  $\pi_g$ , initialised as  $\pi_g = \pi_e$  on  $\mathcal{D}$  using behavioural cloning. We note this procedure can be extended to alternative imitation learning algorithms (such as DAgger) provided  $f$  can be inferred during generation of source windows  $x^{(c,i)}$ . Under the assumption that the environment satisfies the Markov property, the policy learning process is unaffected by the windowing procedure. As the full trajectories do not need to be reconstructed, limitations of other methods such as requirement for blending or enforcing temporal consistency are inapplicable to this work<sup>28</sup>. Furthermore, as each trajectory is decomposed into  $T - N + 1$  windows, the windowing approach has the effect of significantly augmenting the training data.

### Style transfer

In this work, we consider neural style transfer<sup>25</sup> as a translation function wherein  $x^{(g)*}$  arise from solving the style transfer optimisation problem:

$$x^{(g,i)*} = \arg \min_{x^{(g,i)}} \left( w_c L_c(x, x^{(c,i)}) + w_s L_s(x, x^{(s,j)}) \right) \quad (9)$$

where  $w_c$  and  $w_s$  are the content and style weights, respectively and  $L_c, L_s$  are content and style loss contributions respectively. The content loss is defined as:

$$L_c = \sum_l \sum_{i,j} \frac{1}{2N_l} \left( F_{ij}^{(c,l)} - F_{ij}^{(g,l)} \right)^2 \quad (10)$$

where  $F^{(c,l)}, F^{(g,l)}$  are the feature outputs of layer  $l$  for the content and generated output respectively. The style loss similarly is expressed as

$$L_s = \sum_l \sum_{i,j} \frac{1}{4N_l^2 M_l^2} \left( G_{ij}^{(s,l)} - G_{ij}^{(g,l)} \right)^2 \quad (11)$$

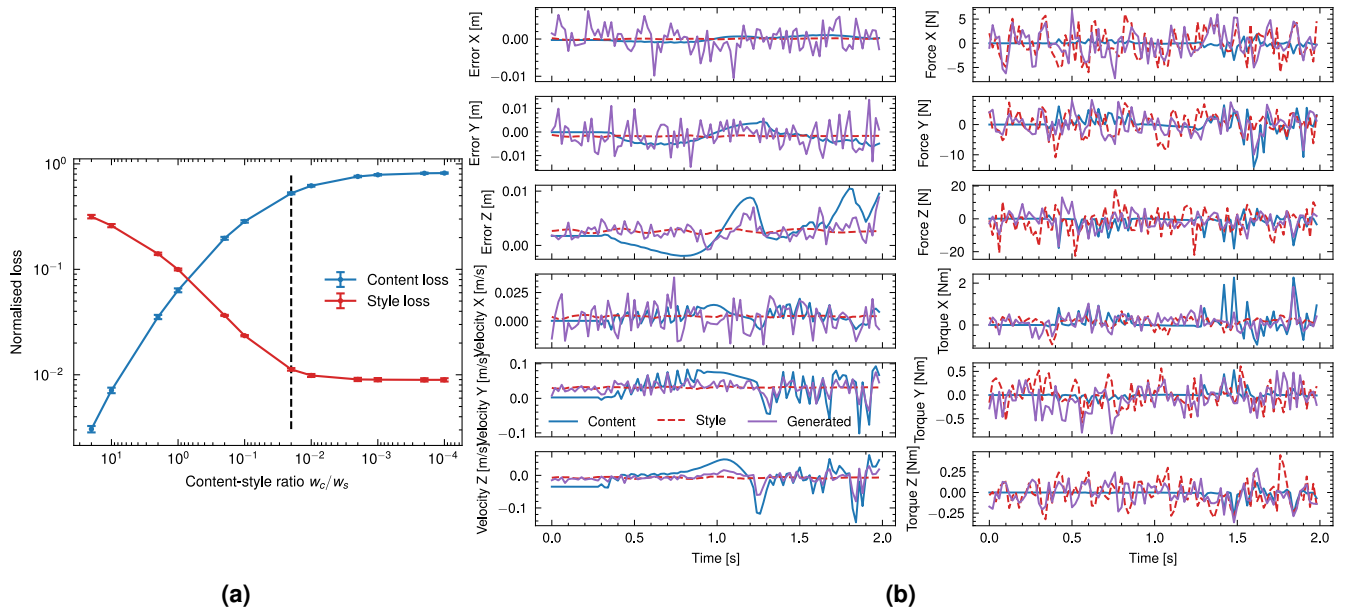
where  $\mathbf{G}^{(s,l)}$  is the style Gram matrix of layer  $l$  outputs  $F^{(s,l)}$

$$\mathbf{G}^{(s,l)} = F^{(s,l)} F^{\top(s,l)} \quad (12)$$

and similarly for  $\mathbf{G}^{(c,l)}$ . The generated windows were initialised as

$$x^{(g,i)} = x^{(c,i)} \quad (13)$$

and (9) optimised by gradient descent using the Adam optimiser. The relative content-style weighting  $w_c/w_s$  was tuned manually through a grid-search procedure. To give insight into this process, Figure 3 shows the effect of the relative content-style weighting ( $w_c/w_s$ ) on their relative (i.e. unweighted) loss contributions and unweighted total loss. At high values of



**Figure 3.** Effect of content-style weight ratio  $w_c/w_s$  on normalised (unweighted) content-style loss, averaged over 5 content-style batches (batch size 256), with chosen value  $w_c/w_s = 0.02$  indicated (dotted line), with comparison of generated and target window (right) with  $w_c/w_s = 0.02$  shown. Decreasing ratio results in diminishing returns on style while diverging substantially from the original content windows. Increasing ratio tends towards identity (generated windows correspond to unaltered simulated windows).

$w_c/w_s$  (shown left), the optimisation weights heavily towards reconstruction of the original content windows, and hence the generated windows do not differ from the simulation dataset in the case of initialisation as (13). With decreasing  $w_c/w_s$  (shown right), the total loss is dominated by increasing content reconstruction error; the generated windows increasingly resemble the real-world dataset, however, do not reflect the simulated task dynamics. Hence,  $w_c/w_s$  was reduced until diminishing returns on the (unweighted) style reconstruction loss was observed. We report relevant optimisation parameters in Table 1. Figure 4 shows the convergence process of the window generation process in (9). Convergence occurs rapidly in the first 200 iterations, with gradual improvement for the remaining 800 iterations, and guides the selection of iteration count in Table 1. As per the initialisation scheme in (13), the content loss increases with each style transfer iteration as the generated window is modified to reduce the style loss.

A compelling advantage of encoder or classifier-based approaches is that they operate on unpaired cross-domain datasets. To improve the realism of generated trajectories, we employ a pairing mechanism that takes advantage of the unsupervised feature representations learned from the source domain data to generate weakly paired content and style windows. An intuitive analogue would be matching images with similar composition and subjects, reminiscent of the weak pairing mechanism in<sup>23</sup>. In the first stage, the real world dataset is encoded in entirety by the encoder network to generate a dataset of embeddings. In the second stage, the simulated content window(s) are encoded and a content-style pairing matrix is constructed by the pairwise cosine similarity between  $x^{(c,i)}$ ,  $x^{(s,j)}$  representations as

$$S_{ij} = \frac{z_i \cdot z_j}{\|z_i\| \cdot \|z_j\|} \quad (14)$$

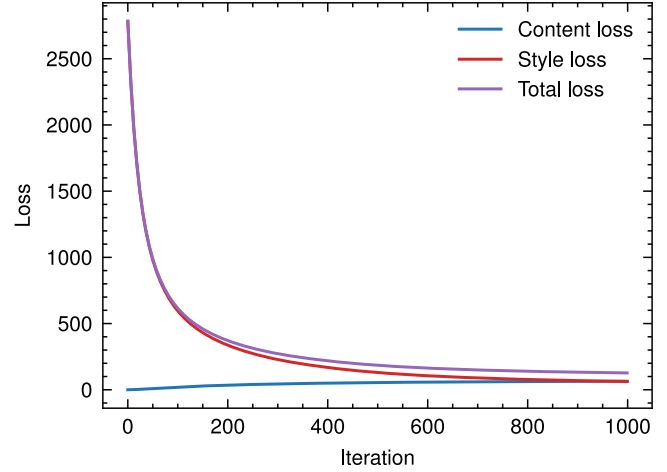
In the last stage, the closest match real embedding is paired with the simulated embedding. For each row  $i$ , the index of the most similar pairing was obtained by:

$$j^*i = \arg \max_j S_{ij} \quad (15)$$

For windows where the pairing diverged substantially from the content, the optimisation process introduced mean shifts into the observations, as well as introducing artefacts from the encoding process. Following the intuition of<sup>26</sup>, qualitatively, we observed that pre-aligning the means of the content and style windows resulted in higher quality generated outputs. Figure 5 shows a representation of the content-style pairings generated by the pairing procedure. The data show the formation of distinct clusters according to simulated and real world trajectories. Unsurprisingly, the real world embeddings with the most matches

**Table 1.** Selected hyperparameters for encoder network and style transfer framework.

Parameter	Value
Encoder learning rate	$1 \times 10^{-3}$
Encoder output channels	[128, 256, 512]
Encoder kernel size	3
Encoder batch size	128
Encoder kernel stride	2
Window size	100
Latent dimensions	130
Content-style ratio $w_c/w_s$	0.02
Style transfer learning rate	0.01
Style transfer iterations	1000
Content layer indices (Fig. 2)	[x]
Style layer indices (Fig. 2)	[2, 5, 7]

**Figure 4.** Convergence plot for style transfer optimisation with parameters from Table 1 for a batch of 256 content-style pairings.

were found predominantly at the intersections of the clusters. This parasitic behaviour is reminiscent of the mode-collapse phenomenon in generative-adversarial networks. Nonetheless, around 50% of real world points were matched at least once, with matched windows dispersed throughout the latents, indicating good coverage of the real world dataset. We evaluated whether the VAE-based matching procedure produced meaningful sim-real pairings by comparing the cosine similarity of matched latent codes to two baselines: (i) the similarity obtained by pairing each simulated window with a randomly chosen real window (random matching of  $j^*$ , as opposed to (15)), and (ii) the similarity between real-window embeddings and latents sampled from the standard normal prior (prior baseline). The mean cosine similarity of the matched pairs was 0.515, compared with 0.269 for random matching, and 0.193 for prior samples. Comparison with Welch's t-test (two-sided) strongly rejects the null that matchings are random ( $t(1479.7) = 34.2$ ,  $t(970.60) = 91.8$ ;  $p < 0.001$ ).

For adaptation, 50 episodic trajectories were collected in source domain with the expert policy, which formed the content dataset. Note this is separate from the 680 trajectories used to train the VAE, which consisted of a mixture of off-policy and on-policy trajectories. For this work, the style dataset consisted of 148 off-policy trajectories collected from the real world. We note this is not a hard requirement; dataset size is motivated primarily by avoiding breakdown of the pairing and style transfer mechanism where content and style windows diverge substantially.

### Experimental setup

As with our previous work, experimental validation was carried out on a KUKA LBR *iiwa* R820 14kg collaborative robot equipped with a wrist-mounted motorised slitting saw tool. The *iiwa* was connected via the Fast Research Interface (FRI) to a Robot Operating System (ROS) workstation with a communication frequency of 500Hz. The workstation used for experimental validation was equivalent to the training workstation used for initial policy training, as specified in the policy adaptation procedure. The robot was equipped with a motorised slitting saw tool; whereas geometric parameters of the tool reflect the training parameters in Table 2, the number of teeth was doubled to introduce further cross-domain mismatch.

The cutting task was represented as a single conventional milling pass over an material with variable geometry, following a nominal trajectory defined at the material surface. As proof of principle, the reference path was defined manually with respect to the surface for all case studies. During the cutting task, the policy provides as output a translational stiffness, incremental offset to the depth of cut (DoC), and the feed rate, relative to the planned (nominal) trajectory. The nominal feed rate was chosen as 0.75 m/min. The controller damping gain  $\mathbf{K}_d$  was adjusted independently according to the stiffness to provide a damping ratio of 1.0 (i.e. critically damped). Trajectory tracking was achieved according to the operational space control law

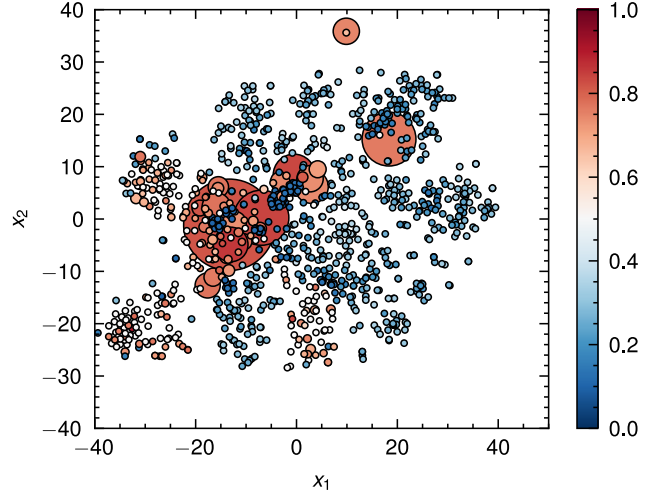
$$\Gamma = \mathbf{J}^T [\hat{\Lambda}(q) (\mathbf{K}_d(t)\dot{e} + \mathbf{K}_p(t)e) + \hat{\mu}(q, \dot{q}) + \hat{\rho}(q)] \quad (16)$$

where  $\Gamma$  are the commanded joint torques,  $\mathbf{J}$  the robot Jacobian, and  $\hat{\Lambda}$ ,  $\hat{\mu}$ ,  $\hat{\rho}$  are the estimated operational space inertia matrix, Coriolis & centrifugal forces, and gravitational forces respectively.

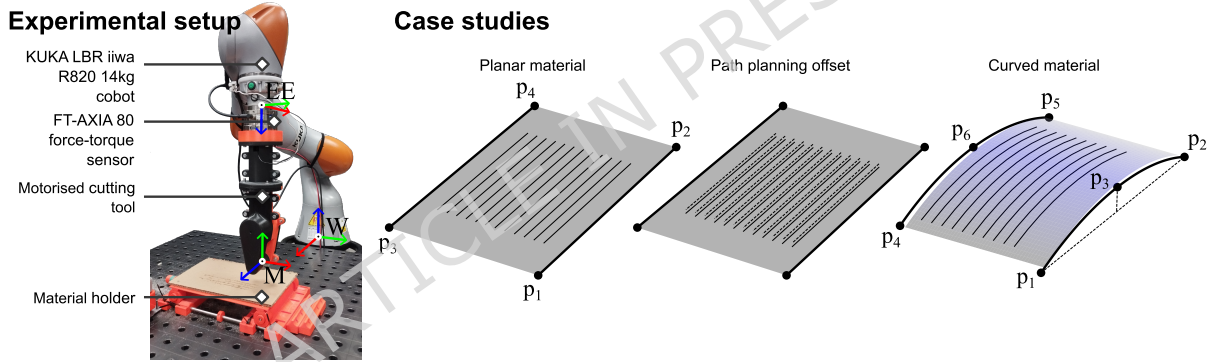
During the cutting task, the process force was monitored via an FT-AXIA 80 force-torque sensor, mounted at the robot wrist. However, our method in principle is applicable to different types of sensors, such as those built in to the *iiwa*, provided real world examples collected with such sensors. Prior to each trial, the force sensor was biased at the start of the trajectory.

**Table 2.** Table of model parameters for cutting simulation (source domain)

Parameter	Value
Pitch angle [rad]	0.126
Helix angle [rad]	0.0
Radius [m]	0.025
Cutter width [m]	0.0005
Cutting elements (flutes)	50
Spindle speed [rpm]	1000
Material cutting	
-mechanistic constant ( $K_c$ ) [N/mm <sup>2</sup> ]	<i>variable</i>
Material edge	
-mechanistic constant ( $K_e$ ) [N/mm]	<i>variable</i>



**Figure 5.** t-SNE embedding diagram of content-style pairings. The points are coloured according to their class (simulation, blue / real world, red) with intensity according to the cosine similarity of their closest match, diverging from 0.5. The area of each real world embedding point is directly proportional to the number of times the corresponding window was matched.



**Figure 6.** Overview of the experimental setup used for real world cutting experiments.

Force sensor gravity compensation was achieved via the following correction:

$$F_{ext}^W = \mathbf{R}_{EE}^W F^{EE} + mg (\hat{z} - \mathbf{R}_{EE}^W \mathbf{R}_{W,0}^{EE} \hat{z}) \quad (17)$$

where  $m$  is the tool mass,  $g$  is the gravitational acceleration,  $F_{ext}^W$  is the measured external force in the world frame  $W$ ,  $\hat{z}$  is the z-axis basis vector of  $W$ , and  $\mathbf{R}_W^{EE}$ ,  $\mathbf{R}_{W,0}^{EE}$  are the world to end-effector (EE) rotations at the current end-effector pose, and bias pose, respectively. An overview of the experimental setup and case studies is provided in Figure 6.

## Results

In this section, we evaluate the proposed method in comparison to the unadapted expert policy and state-of-the-art methods based on the previously established experimental setup. To demonstrate the performance of each method on a range of materials, cutting trials were carried out on polyurethane foam, cardboard, corrugated plastic, mica and aluminium. We further establish 3 separate case studies on each material to evaluate the policy performance under different path planning conditions. We evaluate each method by task completion time, average path deviation, average tool load, material removed volume (MRV), and similarity of the adopted action trajectories to the source domain expert actions, averaged over 5 trials per material for each strategy, and aggregated over all materials. To mitigate effects of drift (e.g. tool wear, temperature, calibration errors), trials for each strategy were interleaved.

## Comparison methods

For the subsequent real world experiments, we adopt the following terminology to denote comparison methods: ‘Expert’ refers to the unadapted source simulation expert policy, as transferred directly to the real world task. ‘BC’, or standalone behavioural cloning, represents our previous work, in which the simulation is augmented with a Gaussian process (GP) regression model trained on aligned trajectories from 14 preliminary experiments on aluminium and mica (a subset of the 148 real world trajectories considered in this work). In the case of real world trajectories, the number of trajectories used for the previous method is substantially lower, however, relies heavily on modelling assumptions which are embedded into the GP kernel & alignment approach, which are not used in this work. ‘CVAE’ represents a conditional variational autoencoder using the same real world dataset as adopted for style transfer. Note in this instance, the encoder itself is trained on the entire dataset of both real world and simulation data, conditioned on a one-hot domain label (simulation or real world). Simulated data are encoded as with the style transfer approach, however, at decoding time, the one-hot class label is swapped to generate a synthetic window of the desired class. ‘CycleGAN’ is also introduced as a comparison method. In this instance, the surrogate real world dataset is synthesised by the sim-to-real generator network. With all methods, the generator / encoder architecture was chosen equivalent to Table 1. For CycleGAN, a smaller discriminator network, with output channels [64, 128, 256] was used due to mitigate the well-known ‘vanishing gradient’ problem during GAN training. All other hyperparameters were chosen to be equivalent to the CycleGAN study. All methods were employed with behavioural cloning as per the self-supervision procedure introduced in this work. Additionally, as a benchmark, we include a “baseline” strategy in which the process parameters are held constant at the nominal feed rate (0.75m/min) and depth of cut of 1 mm, applied to all materials.

## Planar material case study

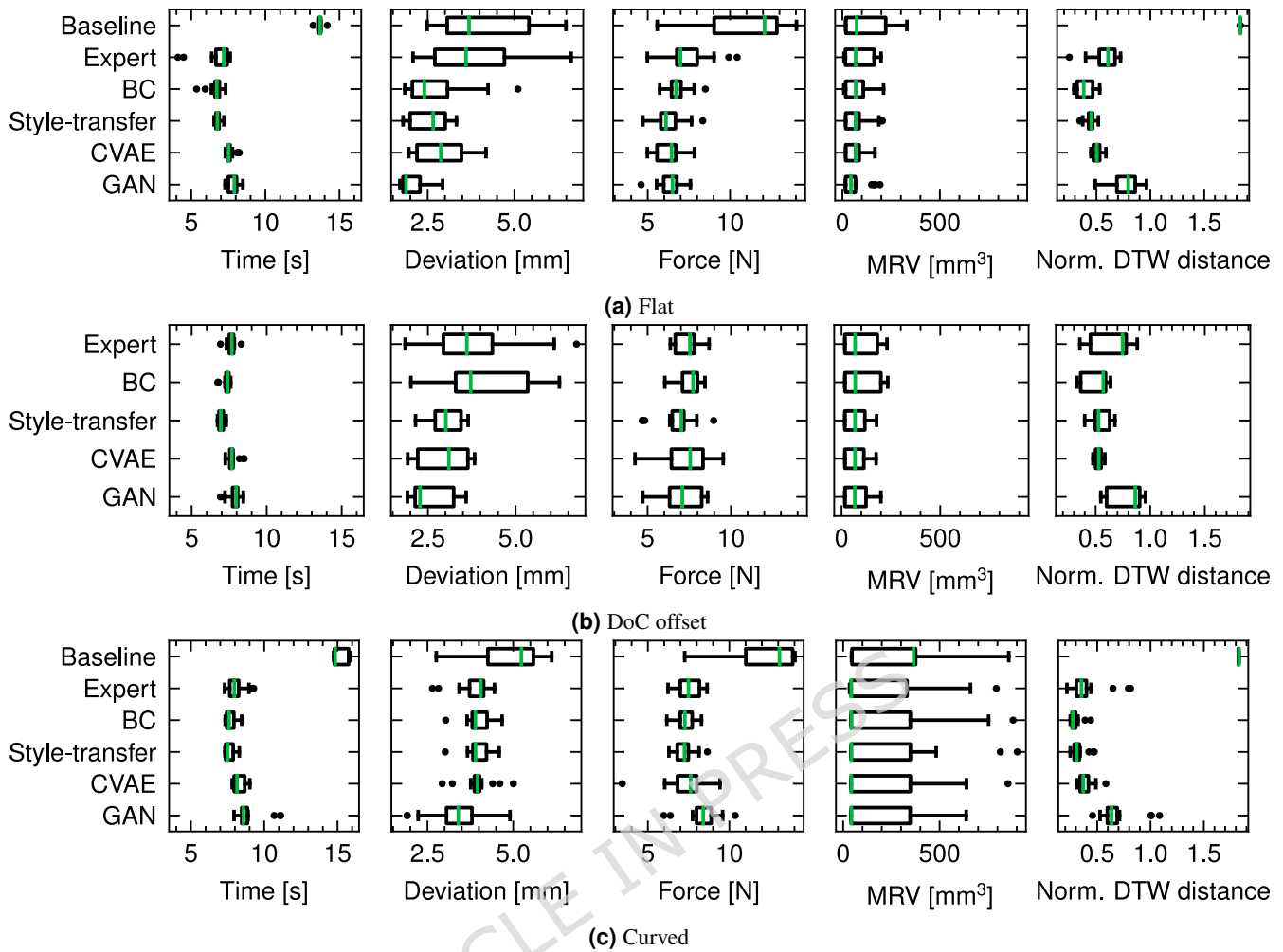
Each strategy was initially tested on a planar material, with the reference path calibrated at the material surface. For the calibration procedure, the surface was modelled as a warped plane interpolated between 4 corner points obtained via guarded move with a force threshold of 1N, with the exception of foam, where contact was confirmed visually. The performance of each strategy for the planar case study is outlined in Figure 7a. To aid interpretation, the significance of the difference in metrics was tested via one-way ANOVA. The normality and homoscedasticity assumptions of ANOVA were tested via the Shapiro-Wilk and Levene methods respectively. A significance level of  $\alpha = 0.05$  was used for all tests. Metrics that did not satisfy the assumptions were transformed via Box-Cox transform:

$$y = \begin{cases} x^\lambda - 1 & \text{if } \lambda \neq 0 \\ \log(x) & \text{otherwise} \end{cases} \quad (18)$$

where  $\lambda$  is chosen to maximise the log-likelihood of the transformed data under a normality assumption. In the case of completion time and average force, the assumptions of ANOVA were satisfied (Shapiro  $p = 0.361$ ,  $p = 0.355$ ; Levene  $p = 0.0689$ ,  $p = 0.0983$ , respectively). Average path deviation and MRV did not satisfy the normality assumption after transformation, and in this case the Kruskal-Wallis test was adopted without transformation. For both task completion time and average force, one-way ANOVA revealed significant effects of strategy on performance ( $F = 61.1$ ,  $p = 1.14 \times 10^{-27}$ ;  $F = 6.74$ ,  $p = 6.52 \times 10^{-5}$  respectively) between strategy and these performance metrics.

To examine the effect of individual strategy on the performance metrics, the Tukey Honestly Significant Difference (HSD) was used for ANOVA, and the Dunn post-hoc test for Kruskal-Wallis. No significant difference in task completion times was found between style transfer and BC, whereas the former outperformed all other methods. Style transfer had the largest effect relative to GAN ( $-1.00$  s) and the smallest relative to the Expert ( $-0.329$  s). For path deviation, style transfer significantly differed from the Expert ( $-1.50$  mm,  $p = 0.000196$ ) and GAN ( $0.451$  mm,  $p = 0.005074$ ) strategies, however, results were inconclusive for BC ( $p = 0.560$ ) and CVAE ( $p = 0.109$ ). Style transfer was further found to significantly outperform the Expert and BC strategies in minimising average force ( $-1.273$  N,  $p = 0.0001$ ;  $-0.651$  N,  $p = 0.0352$ ), however, no significant difference was found between style transfer and the CVAE and GAN strategies ( $p = 0.867$ ,  $p = 0.611$ ). The choice of strategy was found to have no conclusive effect on MRV (Kruskal  $H = 2.87$ ,  $p = 0.578$ ). This result appears surprising in light of the differing action selection apparent for each strategy, particularly in DoC.

To examine the effect of the adaptation methods on the agent actions, the actions taken during each trial were compared with 50 simulated experiments (i.e. source domain) carried out with the source domain expert, and the similarity of action trajectories evaluated by normalised dynamic time warping (DTW) distance. The strategies that adopt actions that are more broadly similar to the source domain expert will score lower on this metric than those that deviate substantially from the expert behaviour. The expert policy itself was included in this comparison since it is being applied to the *target* domain. We report effect sizes as Hedges’  $g$ . Clear differences between the strategies were indicated (Kruskal  $H = 1930$ ,  $p = 0.0$ ), with style transfer yielding large improvements relative to the Expert  $g = 0.875$  and GAN  $g = 2.18$ , a moderate improvement for CVAE  $g = 0.575$  and a small reduction in performance relative to BC  $g = -0.370$ . Post-hoc testing indicated a high significance level in these effects ( $p \leq 3.65 \times 10^{-17}$ ) for all comparisons.



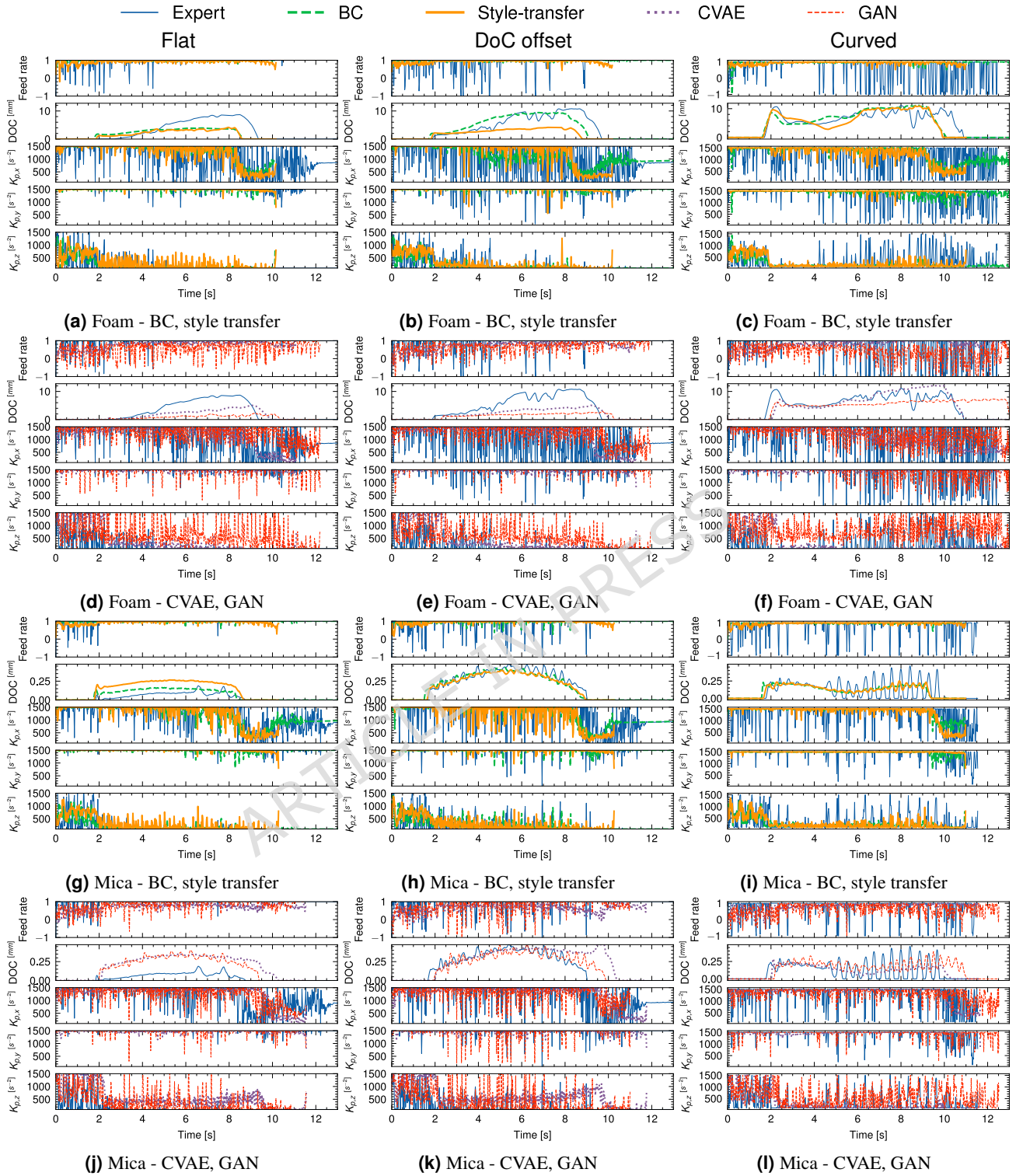
**Figure 7.** Boxplot summary of performance metrics for the style transfer trained policy and comparison methods, aggregated over all materials. Metrics include task completion time, average path deviation, average load force, average (normalised) dynamic time warping (DTW) distance between each strategy and the simulation expert policy (lower better), and material removed volume (MRV, higher better).

To examine the behaviour of each strategy in more detail and enable qualitative comparisons between each strategy, the action trajectories adopted by each policy during an example trial on foam and mica are presented in Figure 8. From Figure 8a, 8d, 8g, 8j, the action trajectories were broadly similar between BC and style transfer across both materials. Style transfer adopts a more correct behaviour of reducing the feed rate prior to engagement with the material, as compared with BC. Conversely, the GAN policy diverges substantially from the expert behaviour which corroborates the DTW metric results. All adapted policies adopted a more consistent DoC throughout both trials than the unadapted expert policy. Differences between the policy behaviour on each material were mainly evident in the DoC behaviour, transverse stiffness ( $K_x$ ) and, to a lesser extent, the normal stiffness ( $K_z$ ).

### Robustness to path planning offset

To examine the robustness of the method to path planning errors - for example, due to errors in calibration, surface position estimation or noise, we further examine the performance of all strategies with a path planning offset of 1mm, inset into the ground truth material surface. The performance of each strategy was evaluated over 3 trials per strategy, per material.

Similarly to the planar case, strategy significantly impacted task completion times (ANOVA  $F = 17.0$ ,  $p = 9.94 \times 10^{-10}$ ); style transfer again differed significantly from all strategies except BC (Tukey HSD  $p = 0.0694$ ), and improvements over other strategies being similar to the planar case study, albeit more consistent across strategies (effect size range  $-0.658$ – $-0.850$  s). Whereas path deviation was also influenced by strategy (ANOVA  $F = 4.61$ ,  $p = 0.00235$ ), post-hoc testing indicated only GAN differed significantly from the BC ( $p = 0.0047$ ) and Expert ( $p = 0.0125$ ) strategies. Although group means were more



**Figure 8.** Comparison of agent actions for foam and mica for planar, DoC offset and curved case studies respectively. Actions include the relative to nominal feed rate adjustment, with 0 corresponding to no change and 1 to double the nominal feed rate, depth of cut (DoC), and controller stiffness in transverse, feed direction and normal directions respectively ( $K_{p,x}$ ,  $K_{p,y}$ ,  $K_{p,z}$ . Units of  $K_p$  are chosen consistent with (16)).

concentrated than in the planar case, path deviation was notably more consistent across trials for style transfer, CVAE, and GAN, implying these strategies were better able to tolerate the path planning offset and maintain stable path tracking across materials. MRV was again unaffected by strategy (Kruskal-Wallis  $H = 2.61$ ,  $p = 0.624$ ), and contrasting the planar case study, no significant differences were observed in average tool load (ANOVA  $F = 1.06$ ,  $p = 0.382$ ). Similarly to the planar case study, there was a clear separation between the strategies in terms of similarity to expert actions (Kruskal  $H = 688$ ,  $p = 9.33 \times 10^{-148}$ ). Post-hoc testing indicated style transfer was distinct from the comparison methods, with the least significant result being with CVAE ( $p = 0.0314$ ), small negative effects for BC  $g = -0.321$  and CVAE  $g = -0.151$  and positive effects relative to Expert  $g = 0.682$  and GAN  $g = 1.31$  strategies.

Figure 8b, 8c, 8h, 8k shows the agent actions for the offset case study. All strategies exhibited a more sporadic DoC behaviour than the planar case study, with style transfer exhibiting the most consistent DoC behaviour across both materials, and matching more closely to the planar case study behaviour, supporting observations regarding the consistency of the path deviation. All strategies exhibited a more aggressive variation in stiffness relative to the planar case study, indicating a compensatory response to the offset cutting depth.

### Non-planar surfaces

We further showcase the performance of each strategy when both material and surface geometry are altered to varying degrees of curvature. Consistent with the planar case study, the reference path with respect to the surface was assumed already known; however, we note that numerous path-planning methods have been proposed in the context of milling, including the case where surface geometry is unknown<sup>3</sup>. For this case study, we assume the material is a thin plate under pure bending, with the surface modelled as a section of a truncated oblique cone – in other words, an interpolation between two circular arcs. The arc parameters for each endpoint were derived from a 3-point estimation obtained similarly to the planar case study. Curvatures ranged between  $2.36 \text{ m}^{-1}$  and  $4.04 \text{ m}^{-1}$  across materials. Cardboard was excluded from the set of materials since the maximum curvature generated during preliminary experiments did not meaningfully differ from the previous case studies.

As with the prior case studies, strategy had a significant effect on completion time (Kruskal  $H = 38.5$ ,  $p = 8.44 \times 10^{-8}$ ) and in post-hoc testing, style transfer outperformed all strategies except BC ( $p = 0.529$ ). The effect of style transfer largely reflected the planar case study, with  $-1.00$  s relative to GAN, and  $-0.413$  s relative to the Expert. Differences in path deviation were inconclusive compared to the planar case study, (Kruskal  $H = 9.77$ ,  $p = 0.0445$ ) with the most significant result from post-hoc testing arising between GAN and style transfer ( $p = 0.0589$ ); however, differences in average force were more pronounced (ANOVA  $F = 7.71$ ,  $p = 0.000025$ ), with style transfer significantly outperforming GAN ( $-1.25$  N,  $p = 0.0001$ ) but not the other strategies. Corroborating the previous case studies, MRV did not significantly differ between strategies (Kruskal  $H = 2.15$ ,  $p = 0.708$ ). Furthermore, action similarity again revealed clear separation between strategies (Kruskal  $H = 1390$ ,  $p = 2.64 \times 10^{-300}$ ), with style transfer exhibiting the largest deviation from GAN ( $g = 2.14$ ) and significant differences from all others (BC  $g = -0.264$ , CVAE  $g = 0.503$ , Expert  $g = 0.487$ ).

The agent actions, as shown in Figure 8c, 8f, 8i, 8l, show similar behaviours to the offset case study, with differences in DoC behaviour becoming more pronounced, particularly for the expert policy. CycleGAN adopted a highly sporadic action profile in feed rate and stiffness, particularly for the foam trials. A hypothesis for this behaviour is that the curved material presents a more challenging case for the agent and the much lower cutting forces limit information available to the agent to make decisions. Therefore, the actions resemble those at the beginning of the planar trials in which the agent is in free space and has no information about the contact state or tool engagement. Style transfer and BC both exhibited less consistent DoC behaviour than the planar case studies on foam, however, produced smoother action trajectories that were strongly correlated to the engagement state - for example, contact initiation was well-demarcated for both strategies.

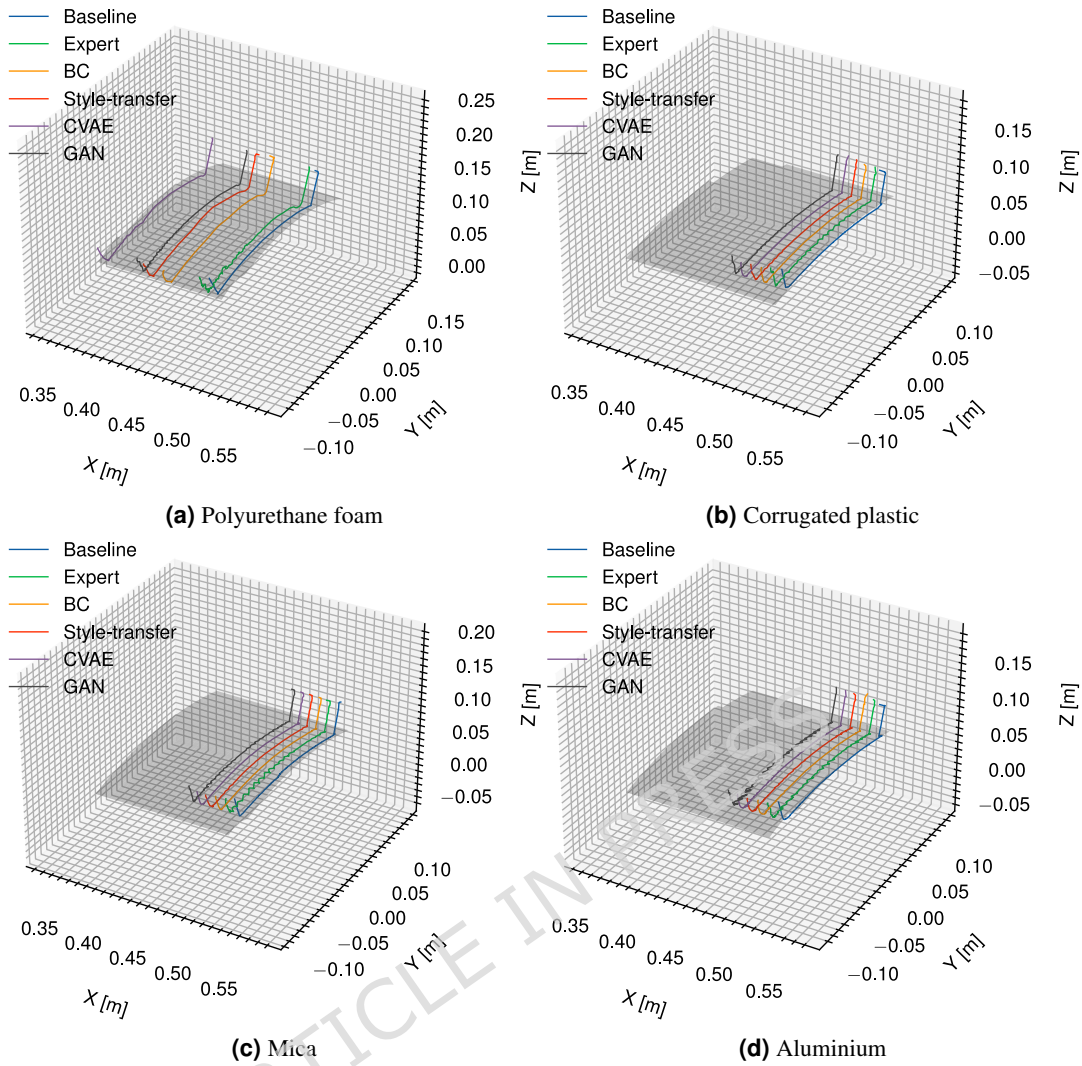
### Qualitative comparison

Figure 9 shows a representative example of the 3D TCP positions adopted by each strategy for a single cutting trial. The TCP trajectories adopted exhibited clear defects for the expert and GAN trials, which were evident across both low and high stiffness materials. On the low stiffness materials, such as in Figure 9a these were evident as low-frequency irregularities, resembling a random walk, whereas for the high stiffness materials, this was exhibited as a higher frequency “wobble”, which were unrelated to known phenomena such as chattering. These defects were suppressed or entirely absent during the BC, CVAE and style transfer trials, with these methods yielding similar qualitative improvements across all materials.

To explore this in greater detail, we employ the smoothness measure proposed in<sup>33</sup>, which has been employed in previous work on sim-to-real transfer<sup>34</sup> to evaluate quality of action trajectories:

$$S_m = \frac{2}{nf_s} \sum_i^n M_i f_i \quad (19)$$

where  $n$  is the number of frequency bands,  $f_s$  is the action sampling frequency and  $M_i$ ,  $f_i$  are the amplitude and frequency of the  $i$ th band, respectively. Frequency bands were obtained by transforming the action trajectories to frequency domain using the



**Figure 9.** 3D plot of TCP paths adopted by each strategy with respect to the material surface - qualitative defects are shown in the “expert” and “GAN” strategies, which exhibit transverse path deviations on the stiffer materials.

discrete Fourier transform. High frequency oscillations in the action trajectories are undesirable when deploying to a physical robot setup due to increased power consumption, wear and reduction in control performance<sup>33</sup>. Correspondingly, higher values of  $S_m$  correspond to stronger high-frequency oscillations in the action trajectory - lower values of  $S_m$  therefore imply smoother action trajectories. Table 3 illustrates the smoothness measure for each action component as illustrated graphically in Figure 8. Across all three case studies, the style-transfer policy demonstrated consistently better action smoothness than the GAN baseline and unadapted expert policy, but did not match the performance of the CVAE and GP-based adaptation methods.

## Discussion

For the cutting task, the proposed method was evaluated based on task completion times, average path deviation, tool load (average force), material removed volume, behavioural similarity to expert action trajectories in source domain, and qualitatively by the action trajectories, ability to maintain consistent cutting conditions (e.g. depth of cut), as well as TCP trajectories. Relative to the comparison methods – consisting of the unadapted source domain expert policy (Expert), our previous work (BC), conditional variational autoencoder (CVAE) and CycleGAN (generative adversarial network) – the proposed method based on style transfer consistently achieved significant reductions in task completion time across all case studies. Compared to BC and CVAE, style transfer showed comparable performance but did not uniformly surpass them across all metrics.

The reduced influence of strategy in the offset path case study is consistent with the constraint imposed by inserting the path into the material, which limits the ability of the agent to regulate the true DoC. It also implies a common limitation of these

**Table 3.** Comparison of action smoothness across strategies and case studies. Lower values indicate overall smoother and less aggressive action profiles.

Case study	Strategy	Feed rate	DoC	Stiffness ( $K_p$ )
Planar	Expert	0.0110	$1.02 \times 10^{-6}$	10.7
	BC	0.00202	$6.93 \times 10^{-7}$	3.30
	CVAE	0.00489	$6.04 \times 10^{-7}$	3.90
	GAN	0.0128	$8.96 \times 10^{-7}$	12.2
	Style-transfer	0.00240	$8.19 \times 10^{-7}$	6.39
DoC offset	Expert	0.0160	$1.96 \times 10^{-6}$	13.6
	BC	0.00433	$1.13 \times 10^{-6}$	4.96
	CVAE	0.00568	$7.46 \times 10^{-7}$	4.28
	GAN	0.0123	$1.36 \times 10^{-6}$	12.4
	Style-transfer	0.00495	$1.29 \times 10^{-6}$	10.1
Curved	Expert	0.0136	$1.94 \times 10^{-6}$	9.59
	BC	0.00180	$1.08 \times 10^{-6}$	2.49
	CVAE	0.00374	$9.71 \times 10^{-7}$	2.84
	GAN	0.0115	$1.34 \times 10^{-6}$	9.78
	Style-transfer	0.00181	$1.15 \times 10^{-6}$	4.28

methods in modelling out-of-distribution task conditions, wherein offsetting the reference path and nominal feed rate introduces concept shift in the optimal actions across domains in addition to covariate shift in the observations. Although path deviation was more consistent across style transfer, CVAE and GAN strategies than for BC and the expert policy, overall improvements were primarily inconclusive. It is plausible that the inconclusive effects may be attributable to the reduced number of samples for the offset case study.

Qualitatively, the style transfer trained policy demonstrated improved behavioural stability relative to the expert and GAN-trained policies, with smoother action trajectories and more consistent control of depth-of-cut and stiffness, which was robust to perturbations in surface geometry and cutting path. The irregular path deviations observed in the TCP trajectories were attributable to the largely sporadic action trajectories of the expert policy, and, to a lesser extent, the GAN strategy. For the stiffer materials, deviations in the path are caused by contact instabilities resulting from interaction between the policy stiffness and the environment stiffness. These behaviours were largely absent with the BC, CVAE and style transfer strategies. Comparison with style transfer by smoothness of action trajectories corroborates these observed qualitative improvements over GAN-based adaptation, but did not consistently reach the smoothness achieved by BC or CVAE methods. Nevertheless, comparison of these methods by average DTW distance from the expert actions in source domain indicated the expert policy behaviour was better preserved during real-world experiments than with the CVAE method for the planar and curved material case studies. This indicates that style-transfer provides a viable mechanism for sim-to-real domain adaptation without markedly degrading control quality. We hypothesise that the poorer performance of CycleGAN-based domain adaptation arises from its limited capacity to preserve task-relevant structure in the translated observations, which has been documented in related work<sup>35</sup>. While CycleGAN has been effective in visual domains where semantic content remains invariant under style changes—e.g. image-to-image translation, its application to time-series control tasks may disrupt temporal dependencies or distort dynamics-critical features, leading to degraded policy performance.

A further advantage of the proposed approach is that the real world data are not required during the process of training the VAE feature extraction model, and are only included at generation time in (9). This contrasts the CVAE and CycleGAN methods, which rely on these datasets to learn conditional mappings to the embedding space in the case of the former, and sim-real translation networks in the case of the latter. Therefore, the proposed method could be employed with different real world datasets (e.g. adaptation to different sensors and tools), incurring only the overhead of adapting the RL policy itself. Complex re-training of the CVAE or GAN models - with associated challenges - is therefore avoided.

## Conclusion

An example-based approach for sim-to-real transfer in robotic control was proposed based on the principle of neural style transfer. Empirical results on a robotic cutting task demonstrate that the proposed method achieves comparable or superior performance to our previous work, conditional variational autoencoders, and CycleGAN-based time series translation across diverse materials and geometric scenarios, while substantially relaxing the assumptions of our previous example-based work.

The proposed method is sample-efficient, demonstrated with 148 off-policy real world trajectories versus 32000 for initial policy training, and avoids the need for training domain discriminator, generator or corrective models, a crucial limitation of previously proposed adaptation methods.

We note the limitation that this work does not explicitly address differing cross-domain target (action) distributions or compatibility of generated trajectories with robot kinematic and dynamic constraints. We posit such constraints could be formulated as part of the optimisation process wherein physical feasibility losses are jointly optimised with style and content losses, and represents a possible extension of this work. Additionally, the quality of generated trajectories and pairings is expected to deteriorate with low coverage of real-world examples, weak content-style match similarity, or parasitic matching where a small subset of real trajectories dominate the pairing.

## Data availability

The datasets generated during and/or analysed during the current study are available in the Figshare repository, DOI 10.6084/m9.figshare.28983659 [FOR REVIEW PURPOSES ONLY: <https://figshare.com/s/23c98a80f2f306474f83>].

## Funding Declaration

This work was supported by the UK Research and Innovation (UKRI) project “Research and Development of a Highly Automated and Safe Streamlined Process for Increase Lithium-ion Battery Repurposing and Recycling” (REBELION) under Grant 10079049.

## Acknowledgements

The authors would further like to acknowledge Abdelaziz Wasfy Shaarawy, Carl Meggs and Christopher Gell respectively for assistance with experimental validation, design of material holder and cutter tool for experiments herein.

## Author contributions

Conceptualisation - A.R. and J.H.; data curation - J.H.; formal analysis - J.H.; funding acquisition - A.R. and R.S.; investigation - J.H.; methodology - J.H. and A.R.; project administration - A.R. and R.S.; software - J.H.; resources - J.H., A.R. and R.S.; supervision - A.R. and R.S.; validation - J.H. and A.R.; visualisation - J.H.; writing (original draft) - J.H.; writing (review & editing) - J.H. and A.R. and R.S.

## Competing interests

The authors declare no competing interests.

## References

1. Vongbunpong, S., Kara, S. & Pagnucco, M. Basic behaviour control of the vision-based cognitive robotic disassembly automation. *Assembly Autom.* **33**, 38–56, DOI: [10.1108/01445151311294694](https://doi.org/10.1108/01445151311294694) (2013).
2. Vongbunpong, S., Kara, S. & Pagnucco, M. Learning and revision in cognitive robotics disassembly automation. *Robotics Comput. Manuf.* **34**, 79–94, DOI: [10.1016/j.rcim.2014.11.003](https://doi.org/10.1016/j.rcim.2014.11.003) (2015).
3. Gao, Y., Gao, H., Bai, K., Li, M. & Dong, W. A robotic milling system based on 3d point cloud. *Machines* **9**, DOI: [10.3390/machines9120355](https://doi.org/10.3390/machines9120355) (2021).
4. Jiang, Y. *et al.* Contour error modeling and compensation of cnc machining based on deep learning and reinforcement learning. *The Int. J. Adv. Manuf. Technol.* **118**, 551–570, DOI: [10.1007/s00170-021-07895-6](https://doi.org/10.1007/s00170-021-07895-6) (2022).
5. Lu, Y. *et al.* A novel disassembly process of end-of-life lithium-ion batteries enhanced by online sensing and machine learning techniques. *J. Intell. Manuf.* **34**, 2463–2475, DOI: [10.1007/s10845-022-01936-x](https://doi.org/10.1007/s10845-022-01936-x) (2023).
6. Beltran-Hernandez, C. C., Petit, D., Ramirez-Alpizar, I. G. & Harada, K. Variable compliance control for robotic peg-in-hole assembly: A deep-reinforcement-learning approach. *Appl. Sci.* **10**, DOI: [10.3390/app10196923](https://doi.org/10.3390/app10196923) (2020).
7. Martín-Martín, R. *et al.* Variable impedance control in end-effector space. an action space for reinforcement learning in contact rich tasks. In *Proceedings of the International Conference of Intelligent Robots and Systems (IROS)* (2019).
8. Zhao, Y., Liu, C., Zhiwei, Z., Tang, K. & He, D. Reinforcement learning method for machining deformation control based on meta-invariant feature space. *Vis. computing for industry, biomedicine, art* **5**, 27, DOI: [10.1186/s42492-022-00123-2](https://doi.org/10.1186/s42492-022-00123-2) (2022).

9. Hathaway, J., Rastegarpanah, A. & Stolkin, R. Learning robotic milling strategies based on passive variable operational space interaction control. *IEEE Transactions on Autom. Sci. Eng.* 1–14, DOI: [10.1109/TASE.2023.3279718](https://doi.org/10.1109/TASE.2023.3279718) (2023).
10. Xing, J. *et al.* Domain adaptation in reinforcement learning via latent unified state representation. *CoRR* **abs/2102.05714** (2021). [2102.05714](https://arxiv.org/abs/2102.05714).
11. Li, K. *et al.* A novel adversarial domain adaptation transfer learning method for tool wear state prediction. *Knowledge-Based Syst.* **254**, 109537, DOI: <https://doi.org/10.1016/j.knosys.2022.109537> (2022).
12. Ghifary, M., Kleijn, W. B., Zhang, M., Balduzzi, D. & Li, W. Deep reconstruction-classification networks for unsupervised domain adaptation. In Leibe, B., Matas, J., Sebe, N. & Welling, M. (eds.) *Computer Vision – ECCV 2016*, 597–613 (Springer International Publishing, Cham, 2016).
13. Zhu, T., Ren, R., Li, Y. & Liu, W. A model-based reinforcement learning method with conditional variational auto-encoder. *J. Data Sci. Intell. Syst.* DOI: [10.47852/bonviewJDSIS42022432](https://doi.org/10.47852/bonviewJDSIS42022432) (2024).
14. Takahei, K., Suzuki, N. & Shamoto, E. Identification of the model parameter for milling process simulation with sensor-integrated disturbance observer. *Precis. Eng.* **78**, 146–162, DOI: <https://doi.org/10.1016/j.precisioneng.2022.07.013> (2022).
15. Golemo, F., Taiga, A. A., Courville, A. & Oudeyer, P.-Y. Sim-to-real transfer with neural-augmented robot simulation. In Billard, A., Dragan, A., Peters, J. & Morimoto, J. (eds.) *Proceedings of The 2nd Conference on Robot Learning*, vol. 87 of *Proceedings of Machine Learning Research*, 817–828 (PMLR, 2018).
16. Christiano, P. F. *et al.* Transfer from simulation to real world through learning deep inverse dynamics model. *CoRR* **abs/1610.03518** (2016). [1610.03518](https://arxiv.org/abs/1610.03518).
17. Jung, H. & Oh, S. Gaussian process and disturbance observer based control for disturbance rejection. In *2022 IEEE 17th International Conference on Advanced Motion Control (AMC)*, 94–99, DOI: [10.1109/AMC51637.2022.9729320](https://doi.org/10.1109/AMC51637.2022.9729320) (2022).
18. Wang, K., Ma, J., Man, K. L., Huang, K. & Huang, X. Sim-to-real transfer with domain randomization for maximum power point estimation of photovoltaic systems. In *2021 IEEE International Conference on Environment and Electrical Engineering and 2021 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)*, 1–4, DOI: [10.1109/EEEIC/ICPSEurope51590.2021.9584526](https://doi.org/10.1109/EEEIC/ICPSEurope51590.2021.9584526) (2021).
19. Hathaway, J., Stolkin, R. & Rastegarpanah, A. Imitation learning for sim-to-real adaptation of robotic cutting policies based on residual gaussian process disturbance force model. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2899–2906, DOI: [10.1109/IROS58592.2024.10802660](https://doi.org/10.1109/IROS58592.2024.10802660) (2024).
20. Chou, C.-B. & Lee, C.-H. Generative neural network-based online domain adaptation (GNN-ODA) approach for incomplete target domain data. *IEEE Transactions on Instrumentation Meas.* **72**, 1–10, DOI: [10.1109/TIM.2023.3246495](https://doi.org/10.1109/TIM.2023.3246495) (2023).
21. Zhang, D., Fan, W., Lloyd, J., Yang, C. & Lepora, N. F. One-shot domain-adaptive imitation learning via progressive learning applied to robotic pouring. *IEEE Transactions on Autom. Sci. Eng.* 1–14, DOI: [10.1109/TASE.2022.3220728](https://doi.org/10.1109/TASE.2022.3220728) (2022).
22. Zhu, J.-Y., Park, T., Isola, P. & Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on* (2017).
23. Ikeda, T. *et al.* Sim2real instance-level style transfer for 6d pose estimation. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3225–3232, DOI: [10.1109/IROS47612.2022.9981878](https://doi.org/10.1109/IROS47612.2022.9981878) (2022).
24. Wang, Q. & Breckon, T. P. Generalized zero-shot domain adaptation via coupled conditional variational autoencoders. *Neural Networks* **163**, 40–52, DOI: <https://doi.org/10.1016/j.neunet.2023.03.033> (2023).
25. Gatys, L., Ecker, A. & Bethge, M. A neural algorithm of artistic style. *arXiv* DOI: [10.1167/16.12.326](https://arxiv.org/abs/10.1167/16.12.326) (2015).
26. Gatys, L. A., Ecker, A. S., Bethge, M., Hertzmann, A. & Shechtman, E. Controlling Perceptual Factors in Neural Style Transfer . In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3730–3738, DOI: [10.1109/CVPR.2017.397](https://doi.org/10.1109/CVPR.2017.397) (IEEE Computer Society, Los Alamitos, CA, USA, 2017).
27. Fernandez-Fernandez, R., Victores, J. G., Gago, J. J., Estevez, D. & Balaguer, C. Neural policy style transfer. *Cogn. Syst. Res.* **72**, 23–32, DOI: <https://doi.org/10.1016/j.cogsys.2021.11.003> (2022).
28. Fernandez-Fernandez, R., Aggravi, M., Giordano, P. R., Victores, J. G. & Pacchierotti, C. Neural style transfer with twin-delayed DDPG for shared control of robotic manipulators. In *2022 International Conference on Robotics and Automation (ICRA)*, 4073–4079, DOI: [10.1109/ICRA46639.2022.9812245](https://doi.org/10.1109/ICRA46639.2022.9812245) (2022).

29. El-Laham, Y. & Vyetrenko, S. Styletime: Style transfer for synthetic time series generation. In *Proceedings of the Third ACM International Conference on AI in Finance, ICAIF '22*, 489–496, DOI: [10.1145/3533271.3561772](https://doi.org/10.1145/3533271.3561772) (Association for Computing Machinery, New York, NY, USA, 2022).
30. Chen, B. *et al.* Towards the generalization of time series classification: A feature-level style transfer and multi-source transfer learning perspective. *Knowledge-Based Syst.* **299**, 112057, DOI: <https://doi.org/10.1016/j.knosys.2024.112057> (2024).
31. Long, M., CAO, Z., Wang, J. & Jordan, M. I. Conditional adversarial domain adaptation. In Bengio, S. *et al.* (eds.) *Advances in Neural Information Processing Systems*, vol. 31 (Curran Associates, Inc., 2018).
32. Hathaway, J., Rastegarpanah, A. & Stolkin, R. Learning robotic milling strategies based on passive variable operational space interaction control. *IEEE Transactions on Autom. Sci. Eng.* DOI: [10.1109/TASE.2023.3279718](https://doi.org/10.1109/TASE.2023.3279718) (2023).
33. Mysore, S., Mabsout, B., Mancuso, R. & Saenko, K. Regularizing action policies for smooth control with reinforcement learning. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 1810–1816, DOI: [10.1109/ICRA48506.2021.9561138](https://doi.org/10.1109/ICRA48506.2021.9561138) (IEEE Press, 2021).
34. Christmann, G., Luo, Y.-S., Mandala, H. & Chen, W.-C. Benchmarking smoothness and reducing high-frequency oscillations in continuous control policies. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 627–634, DOI: [10.1109/IROS58592.2024.10802057](https://doi.org/10.1109/IROS58592.2024.10802057) (2024).
35. Arnout, H., Bronner, J., Kehrer, J. & Runkler, T. Dr-tist: Disentangled representation for time series translation across application domains. In *2020 International Joint Conference on Neural Networks (IJCNN)*, 1–8, DOI: [10.1109/IJCNN48605.2020.9207722](https://doi.org/10.1109/IJCNN48605.2020.9207722) (2020).