University of BRISTOL

University of Bristol - Explore Bristol Research
General rights

ARTICLE

Check for updates

Micro-object pose estimation with sim-to-real transfer learning using small dataset

Dandan Zhang [1✉], Antoine Barbot[2], Florent Seichepine[2], Frank P.-W. Lo[2], Wenjia Bai [2], Guang-Zhong Yang[3✉] & Benny Lo[2]

Three-dimensional (3D) pose estimation of micro/nano-objects is essential for the implementation of automatic manipulation in micro/nano-robotic systems. However, out-of-plane pose estimation of a micro/nano-object is challenging, since the images are typically obtained in 2D using a scanning electron microscope (SEM) or an optical microscope (OM). Traditional deep learning based methods require the collection of a large amount of labeled data for model training to estimate the 3D pose of an object from a monocular image. Here we present a sim-to-real learning-to-match approach for 3D pose estimation of micro/nano-objects. Instead of collecting large training datasets, simulated data is generated to enlarge the limited experimental data obtained in practice, while the domain gap between the generated and experimental data is minimized via image translation based on a generative adversarial network (GAN) model. A learning-to-match approach is used to map the generated data and the experimental data to a low-dimensional space with the same data distribution for different pose labels, which ensures effective feature embedding. Combining the labeled data obtained from experiments and simulations, a new training dataset is constructed for robust pose estimation. The proposed method is validated with images from both SEM and OM, facilitating the development of closed-loop control of micro/nano-objects with complex shapes in micro/nano-robotic systems.

[1] University of Bristol, Bristol, UK. [2] Imperial College London, London, UK. [3] Shanghai Jiao Tong University, Shanghai, China. ✉email: ye21623@bristol.ac.uk; gzyang@sjtu.edu.cn

Scanning electron microscopy (SEM), which enables high-resolution imaging of micro/nano-scale objects, is a common tool for micro/nano-robotics development[1]. For example, previous work has demonstrated the use of SEM for assembling[2–4], handling and characterizing nanomaterials[5–8], nanowires, carbon nanotubes, and other nanoscale objects[9,10]. In addition, optical microscope (OM) has been widely integrated with micro-robotic systems. For example, optical micro-manipulation systems integrated with OM have been developed for cell manipulation or other biomedical applications[11]. Therefore, accurate perception of micro/nano-objects has been shown to be essential for closed-loop micro/nano-manipulation and visual servoing, as laboratory-based experiments are often conducted under microscopic observations.

Thus far, most of the micro/nano-scale operations are conducted by an operator using a manual joystick, keyboard[12] or a haptic device[13,14]. To develop a semi- or fully automatic micro/nano-manipulation platform, three-dimensional (3D) pose estimation of the micro/nano-objects is needed, which relies on the microscopic imaging as feedback[15,16]. Previous research has utilized the microscope camera view to estimate the position of micro/nano-objects in 2D with applications in nano-manipulation systems integrated with SEM[17,18], optical tweezers[19] integrated with OM, magnetic microscopic system[20], and atomic force microscopy[21]. However, accurate 3D pose estimation for individual and group-wise robot manipulation has not been fully explored, due to the challenges of 3D pose estimation using monocular microscopic images. Therefore, real-time reliable visual pose estimation of end-effectors and target objects for high-speed micro/nanomanipulation will be the main focus of this paper.

Hitherto, template matching methods have been widely used for the pose estimation of micro/nano-objects. However, the accuracy can be limited as it's difficult to obtain labeled templates for all possible 3D poses. An alternative method is to use simulated images as templates. However, the inaccuracy of micro-fabrication and the characteristics of different image modalities may cause varying appearances of the microrobots in the images obtained from different domains, which is known as the domain gap. The inherent domain gap between the simulated data and the experimental data may induce errors in matching the templates with the real object poses. Feature-based methods, which rely on triangulation with stereo camera views are currently not applicable to microscopic images due to the nature of how the images are acquired. To this end, it is necessary to investigate other new methods for accurate micro/nano-object pose estimation.

Pose estimation for micro/nano-scale systems, such as experimental setups inside SEM, has been investigated based on a geometrical solution[22]. Model-based tracking of magnetic intraocular micro/nanorobots has also been proposed[23]. However, the work mentioned above cannot be used for optical micro/nanorobots due to the transparency of the materials used and the variance of blurriness. Compared to traditional approaches, machine learning based methods can provide more generic solutions for micro/nano-object pose estimation supporting different experimental setups[24–26]. Recent advances in machine learning have offered new opportunities for performing data classification, identification of molecular characteristics[27], consequences prediction and optimal design of materials or nano-devices in nanoscience[28–32]. Recent studies have shown promising results in carrying out accurate predictions even with limited data. Therefore, we aim to investigate machine learning based techniques to assist the perception of micro/nanoscale objects in 3D.

In recent years, artificial neural networks have been investigated for pose estimation of objects at the macroscale level, such as PoseCNN[33], SSD-6D[34], BB8[35], and other methods constructed via deep convolutional neural networks[36]. At the microscale, a CNN-based method for estimating the 3D pose and depth of optically transparent microrobots has been developed[19]. This method relies on a large volume of labeled data of each microrobot with different poses for training, which is expensive due to the high cost of micro/nano-fabrication and difficulty in accurately controlling the pose of the microrobots. To this end, pose estimation of micro/nano-object using a relatively small dataset should be explored to lower the development cost and enable the research in autonomic microrobotic control.

Few-shot learning represents a type of machine learning where the training dataset contains limited labeled data for different classes, contrary to the conventional deep learning which employs a large volume of data for model training[37]. To enable few-shot learning for micro/nano-object pose estimation, labeled data generated in simulation can be used to assist the model training when the experimental data is limited[38]. However, for many tasks, artificial neural network models trained on simulated data do not work well with real experimental data. To bridge the gap between simulated and real data, domain adaptation has been investigated[39]. These include using abstract representations, training invariant feature extractors[40], learning the mapping between feature spaces[41], and image-to-image translation[42,43]. However, some of the aforementioned methods have inherent limitations. For example, abstract representations may not be effective when the image data obtained from different domains have large differences, training invariant feature extractors requires a large dataset, while image-to-image translation may induce artefacts in the images.

To address the limitations mentioned above, a sim-to-real learning-to-match approach is proposed in this paper, which is developed based on the combination of image-to-image translation and training invariant feature extractors. The work presented here is developed based on the few-shot learning concepts, circumventing the need of collecting a large amount of data for model training like most of the existing work[44]. Comparisons are made between the traditional template matching approach and the proposed method for pose estimation of micro/nanorobot based on the image data obtained from various types of image modalities, including SEM and OM images.

Results
The workflow of the proposed method for micro/nano-object pose estimation is illustrated as follows.

(1) Step 1: to reduce the domain gap between the simulated data and the experiment data, a Generative Adversarial Network (GAN) model is developed to learn a mapping from the simulated data to the experimental data, which can translate the labeled images obtained from the source domain (simulation) to the target domain (experiment).

(2) Step 2: to further reduce the discrepancy between the generated data and the experimental data, a feature embedding model is developed for domain adaptation, which minimizes the differences between the images of the micro/nano-objects with the same pose.

(3) Step 3: the embedded domain-invariant features are used to train a multi-layer perception (MLP) model for pose estimation.

(4) Step 4: at test, the pose of the micro/nano-object is predicted online by combining the feature embedding model and the MLP model.

Dataset construction. We assume that the image data obtained from simulation is denoted as S, while the images of the micro/nano-objects collected via experiments are denoted as M. Let

$\mathcal{D}_T^s = \{(x_i^s, y_i^s)\}_i^{N_1}$ denotes a large training dataset made of pairs of simulated data ($\mathcal{D}_T^s \sim \boldsymbol{S}$), where N_1 denotes the number of samples in the simulation domain, x_i^s denotes the image of a micro/nano-object generated from the simulator and y_i^s denotes the corresponding pose value of image x_i^s. Let $\mathcal{D}_T^m = \{(x_i^m, y_i^m)\}_i^{N_2}$ denote a small training dataset made of pairs of data obtained in physical experiments ($\mathcal{D}_T^m \sim \boldsymbol{M}$), where N_2 represents the number of samples in the experimental data domain, x_i^m is the image of a micro/nano-object captured during the physical experiments, y_i^m is the corresponding pose value for image x_i^m.

To reduce the domain gap between $\mathcal{D}_T^s$ and $\mathcal{D}_T^m$, a GAN-based technique is applied for image-to-image translation, which transfers the simulated data to the experimental data domain. This leads to a new dataset $\mathcal{D}_T^{m'} = \{(x_i^{m'}, y_i^{m'})\}_i^{N_1}$ ($\mathcal{D}_T^{m'} \sim \boldsymbol{M'}$), where $\boldsymbol{M'}$ denotes the generated data domain. After the sim-to-real transfer, we assume that the features obtained from $\boldsymbol{M}$ and $\boldsymbol{M'}$ are of similar distributions. The discrepancy between $\boldsymbol{M}$ and $\boldsymbol{M'}$ can be further minimized by training a feature embedding model.

Let θ and γ denote the out-of-plane rotation angle along the X and Y axis respectively. The predictions of angle θ for two microrobots (microrobot A and microrobot B) are used as examples to verify the proposed method in detail. In this case, label $y_i^*(* = s, m)$ is equal to θ in both datasets ($\mathcal{D}_T^s$ and $\mathcal{D}_T^m$). For a more general situation, $y_i^*(* = s, m)$ is a vector constructed by θ and γ, where $\mathbf{y}_i^* = [\theta, \gamma](* = s, m)$. An example about how to estimate the two out-of-plan angles θ and γ simultaneously is introduced in Supplementary Note 5 with experimental verification, while the results are shown in Supplementary Fig. 6.

The definitions of the coordinate and the out-of-plane rotation angle of microrobot A and microrobot B are illustrated in Fig. 1a, b. All the microrobots used for experiments were fabricated using the Two-Photon Polymerization, while the SEM samples were coated with gold using a metal sputtering deposition system (HEX, Korvus Technology) (see Methods section).

For data collection, an SEM (Tescan, Czech) and an OM (Zeiss, UK) were employed to obtain the images of the microrobots with various poses as experimental data (see Methods section). Figure 1c takes microrobot B as an example and demonstrates the images of microrobot B with different out-of-plane poses obtained from the simulator, the SEM and the OT respectively. For the images of the microrobot with the same pose, the domain gaps are significant. Figure 1d, e shows the examples of images obtained from the simulator, the SEM and the OM of microrobot A and microrobot B with the same pose ($\theta = 0°$ is used as an example). For the OM data, the images of the microrobot with the same pose look significantly different, since the images obtained at different depth levels compared to the focal plane of the OM have different levels of blurriness.

For each image collected, it has a corresponding label of θ, which represents the out-of-plane rotation angle along X axis, as shown in Fig. 1a, b. The minimal displacement between the rotation of the microrobots along the X axis is k degree during the experimental data collection process. Let $min(\theta)$ and $max(\theta)$ represent the minimal and maximum out-of-plane rotation along X axis respectively, $min(\theta) = 0°$ and $max(\theta) = 90°$ are used in this paper. Suppose that I represents the number of microrobots required to be fabricated for the data collection, it can be computed based on the following equation:

$$I = [max(\theta) - min(\theta)]/k + 1 \tag{1}$$

For a microrobot printed at a specific pose, K images are collected. N represents the total number of images used to construct the small dataset of the experimental data ($\mathcal{D}_T^m$). Therefore, we have $N = I \times K$ number of images collected in total. The smaller the value of k is, the more precise the pose estimation can be, the more image data we can obtain to construct $\mathcal{D}_T^m$ for model training.

Sim-to-real transfer via GAN. A GAN model can be used for domain adaptation, enabling the sim-to-real transfer[45]. When learning a GAN model, a generator $\boldsymbol{G}$ and a discriminator $\boldsymbol{D}$ are trained in an adversarial manner. In the context of domain adaptation for visual inputs, the generator $\boldsymbol{G}$ takes images from the source domain, and tries to generate output images matching those from the target domain. In the meantime, a discriminator $\boldsymbol{D}$ is trained to distinguish the generated target images and the real experimental images.

Pixel level image translation based on a Pix2Pix model has been developed for image translation[46]. However, paired image data is required, which cannot be applied to sim-to-real transfer since the data from different domains are difficult to pair. CycleGAN[47], DiscoGan[48] and DualGan[49] introduce a cycle-consistent loss to enforce an inverse mapping from the target domain to the source domain in an unsupervised manner, which ensures the translated images can be easily translated back to the original image domain. In this paper, we implement the CycleGAN for image translation, which aims at reducing the domain gap between the simulated data and the experimental data for micro/nano-object pose estimation.

The target of sim-to-real transfer is to learn a mapping function $\boldsymbol{G}(.): \boldsymbol{S} \rightarrow \boldsymbol{M}$, which is known as a generator. Let $\boldsymbol{M'}$ denote the generated images obtained via CycleGAN via $\boldsymbol{M'} = \boldsymbol{G}(\boldsymbol{S})$. We assume that $\boldsymbol{M'}$ has reduced domain gap from $\boldsymbol{M}$. A domain discriminator $\boldsymbol{D_M}$ is used to classify whether a data point is drawn from $\boldsymbol{M}$ or $\boldsymbol{M'}$, which is optimized according to an adversarial loss. Suppose that $\boldsymbol{G'}(.): \boldsymbol{M} \rightarrow \boldsymbol{S}$ is an inverse generator, $\boldsymbol{D_S}$ is a domain discriminator for classifying whether a data point is drawn from $\boldsymbol{S}$ or $\boldsymbol{S'}$.

An overview of the CycleGAN for image data translation between simulated data and experimental data of the microrobots is shown in Fig. 2, where $x_i^{m'} = \boldsymbol{G}(x_i^s)$, $x_i^{s'} = \boldsymbol{G'}(x_i^m)$. To make the generated images indistinguishable from the original images, an adversarial loss is adopted, through which the samples from different domains are not distinguishable after the model training.

In this unpaired image-to-image translation setting, the inverse generator $\boldsymbol{G'}(.)$ is used to map the observations in the target domain back to the source domain ($\boldsymbol{S} \approx \boldsymbol{S'} = \boldsymbol{G'}(\boldsymbol{G}(\boldsymbol{S}))$). A cycle consistency loss $\mathcal{L}_{cyc}(\boldsymbol{G}, \boldsymbol{G'})$ is defined, which is the sum of $\mathcal{L}_{cyc}^1$ and $\mathcal{L}_{cyc}^2$, as indicated in Fig. 2. The cycle consistency loss is used to ensure that the generated images can preserve the content of its original images to some extent. The optimization is formulated as a min-max problem:

$$\boldsymbol{G}^*, \boldsymbol{G'}^* = \arg \min_{\boldsymbol{G},\boldsymbol{G'}} \max_{\boldsymbol{D_S},\boldsymbol{D_T}} \mathcal{L}(\boldsymbol{G}, \boldsymbol{G'}, \boldsymbol{D_S}, \boldsymbol{D_T}) \tag{2}$$

The details of the constructions of loss functions are presented in Methods section. The trained generator $\boldsymbol{G}(.)$ is then applied to translate the labeled simulated images to the generated experimental data, with the pose label simply passed on after translation. After sim-to-real transfer, the learning-to-match approach is employed to capture the domain-invariant features with effective embedding, and ensure precise pose estimation by utilizing a large amount of generated data and the limited labeled experimental data. The learning-to-match approach further reduces the gap between the generated data and the experimental data, which is detailed as follows.
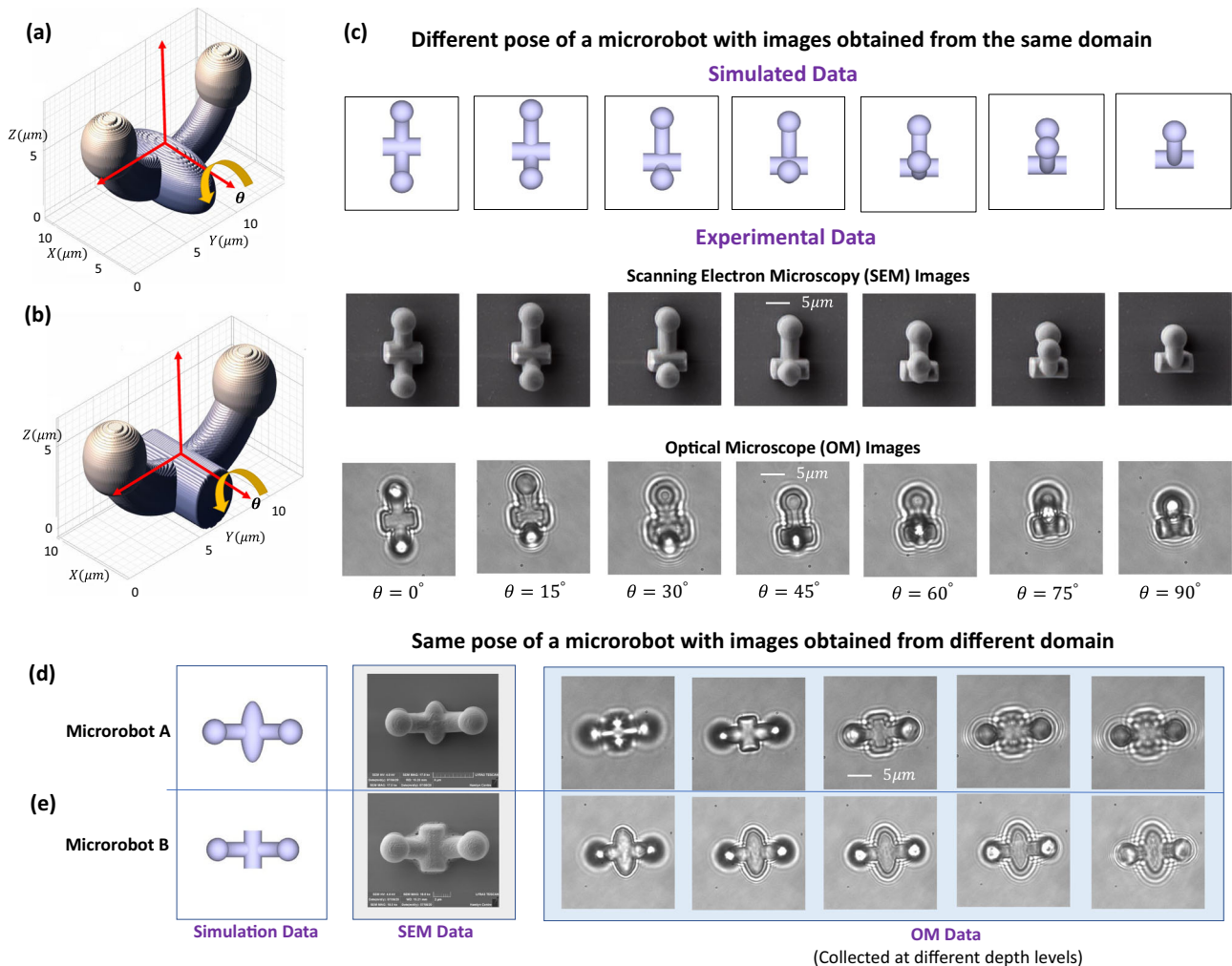
Fig. 1 Overview of the microrobots used for experimental validation. a The definition of the coordinate system and the out-of-plane pose of microrobot A. **b** The definition of the coordinate system and the out-of-plane pose of microrobot B. **c** Different poses of the microrobot B obtained from the simulator, and the images obtained from Scanning Electron Microscopy (SEM) and Optical Microscope (OM). **d** Images obtained from the simulator, the SEM and the OM of microrobot A with the same pose ($\theta = 0°$). **e** Images obtained from the simulator, the SEM and the OM of microrobot B with the same pose ($\theta = 0°$).

As shown in Supplementary Fig. 5, some checkboard patterns can be observed in the generated image, which is the fundamental issue of GAN-based approaches. With the learn-to-match approach described in the next section, the patterns in the background will not influence the pose estimation results. Since the feature embedding model can learn to map images of the microrobots with the same pose to the same location, regardless of the patterns in the background.

Model construction and training. An overview of the learning-to-match approach is demonstrated in Fig. 3. Examples of the generated data obtained via sim-to-real transfer based on Cycle-GAN and the corresponding experimental data with the same pose for microrobot A is shown in Fig. 3a. Figure 3b illustrates the concept of the learning-to-match approach and the overall structure of the pose estimation model. The motivation of the proposed method is to save the computation time by enabling the model to be adapted to new experimental setups quickly. A feature embedding model is trained to project images with different angles to different locations and map images with the same pose angle to the same location in 1D space. Therefore, given a new dataset, the raw image can be compressed while the useful

information of the original image is encoded in the 20-dimensional vector. To enable the precise pose estimation using the new dataset obtained in new environments, we only need to retrain the MLP using the compact features obtained via the feature embedding model for the calibration purpose. In this way, the efficiency of domain adaptation of the proposed method can be ensured.

Let $\mathcal{D}_{\mathcal{T}}^{m'}$ denote the dataset constructed by the images from domain M'. We combine the generated images obtained via GAN (domain transferred simulated data) $\mathcal{D}_{\mathcal{T}}^{m'}$ and the real experimental data $\mathcal{D}_{\mathcal{T}}^{m}$ as $\mathcal{D}_{\mathcal{T}}$. Therefore, $\mathcal{D}_{\mathcal{T}} = \{(x_i, y_i)\}_i^{N_1+N_2}$ consists of images from two domains, i.e. the domain transferred simulated data and the real experimental data with labels. In $\mathcal{D}_{\mathcal{T}}$, we define anchor image data as X^a. Suppose that the pose label of X^a is θ, we select two images (X^p and X^n) randomly among the remaining images in $\mathcal{D}_{\mathcal{T}}$, where the pose label of X^p is θ and the pose label of X^n is θ' ($\theta' \neq \theta$). The target is to train an embedding model $F(.)$ to represent high-dimensional data $X^*(* = a, p, n)$ effectively, where the embedded feature vector is generated by $x^* = F(X^*)(* = a, p, n)$.

The aim is to minimize the distance of embedded features between the anchor image and the positive image of the
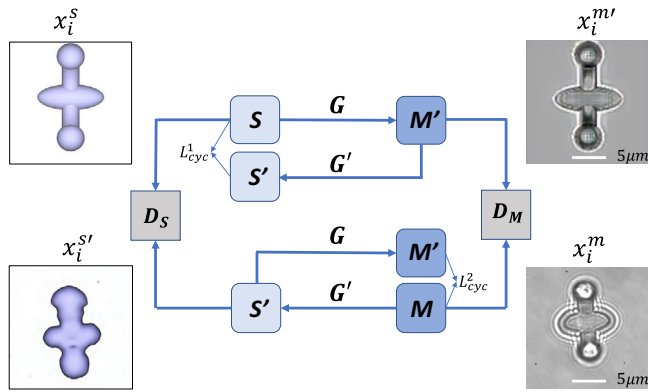
Fig. 2 Image data translation between simulated data and experimental data of microrobots and the illustration of the CycleGAN approach. An image x_i^s obtained from source domain S (simulation domain) is translated to $x_i^{m'}$ in the target domain M', while an image x_i^m obtained from the target domain M (experiment domain) can be translated back to $x_i^{s'}$ in the source domain S'. Microrobot A is used for the demonstration of the sim-to-real transfer approach based on CycleGAN. $G(.)$ is a generator; $G'(.)$ is an inverse generator; $\mathcal{L}_{cyc}^1$ and $\mathcal{L}_{cyc}^2$ represent the cycle consistency loss (see Eq. (7)); D_S and D_M are domain discriminators.

same pose $D_A = \left\| F(X_i^a) - F(X_i^p) \right\|_2^2$, while at the meantime maximize the distance of features between the anchor image and the negative sample $D_B = \left\| F(X_i^a) - F(X_i^n) \right\|_2^2$. Thus, we aim to learn a feature embedding model $F(.)$ such that

$$\phi < L_{embed} = \left\| F(X_i^a) - F(X_i^n) \right\|_2^2 - \left\| F(X_i^a) - F(X_i^p) \right\|_2^2 \\ \forall \left(F(X_i^a), F(X_i^p), F(X_i^n) \right) \in \mathcal{D}_T \quad (3)$$

where ϕ is a margin range. This can be formulated as a triplet loss for model training. Suppose that we sample J frames of X^a and J frames of X^p from $\mathcal{D}_T$ with pose label θ as anchor frames, J frames of X^n from $\mathcal{D}_T$ with pose label $\theta'(\theta' \neq \theta)$ as negative frames.

The embedding function $F(.)$ provides a compact and domain-invariant representation of the microrobot images. This effective feature embedding model can map the images of microrobots with the same pose but from different domains to the same cluster, while the distance between the centers of different clusters is increased, resulting in different clusters are used to represent microrobots with different poses. Subsequently, the embedded feature vector can be fed into the MLP network for pose estimation.

Figure 3b demonstrates the concept of the learning-to-match approach and the overall structure of the pose estimation model. The details of the architecture of the feature embedding model are depicted in Fig. 4a, which includes four convolutional layers and two fully connected layers. The input of the model is the preprocessed image, while the output is an embedded feature vector with the size 1×20.

After training of $F(.)$, the image data $x_i (i = 1, 2, ..., N)$ in $\mathcal{D}_T$ is translated to $x_i' (i = 1, 2, ..., N)$, which forms a new compact dataset $\mathcal{D}_T'$ for model training of pose estimation. The details of the architecture of the MLP neural network model for pose estimation are depicted in Fig. 4b. $\mathcal{D}_T'$ is fed to the MLP neural network model for pose estimation, which is constructed by three fully connected layers, with 128, 64 and 32 neurons respectively. Each fully connected layer is followed by a 'ReLU' activation function before connecting to the next layer. The final fully connected layer is followed by a 'ReLU' activation function and a dropout function to avoid over-fitting, while 'SoftMax' activation

is used to map the feature vector to the target pose value. This MLP model is therefore used for pose value classification. In fact, the MLP model can be easily formulated as a regression model. The main difference comes from the activation function. The 'SoftMax' activation can be changed to a 'linear' activation function if pose value regression is needed. Overview of the MLP neural network model for pose estimation based on regression mode is shown in Supplementary Fig. 2.

Results and analysis

Results for SEM images. For the experimental evaluation, five images are collected for a specific pose of a microrobot ($K = 5$) while k is set as $10°$ ($k = 10$). According to Eq. (1), 10 different classes of microrobots with different pose values are included in the training dataset ($I = 10$), while 50 frames of a microrobot were collected in total to form the small dataset in the domain M ($N = 50$). Examples for sim-to-real transfer based on SEM data can be found in Supplementary Fig. 5.

The training and validation loss of embedding model using SEM data for microrobot A and B in both cases are shown in Fig. 5a, b. For microrobot A and B, the training loss is reduced from the original value of 0.20 and converged to 0.03. As for the validation, the loss value is reduced from 0.20 and converged to 0.01 and 0.02 for microrobot A and B respectively.

We qualitatively evaluate the learned embedding features using t-distributed stochastic neighbor embedding (t-SNE) representations[50,51]. The embedded feature vector has a size of 1×20, while t-SNE can be used to represent the data points in 2D space through non-linear dimensionality reduction in an unsupervised learning manner. The t-SNE based clustering results for the extracted features of microrobots with different pose values are visualized in Fig. 5c, d. It can be seen that the microrobots with different poses can be separated into different clusters with a clear boundary. The visualization of the comparisons between ground truth data and predicted results of SEM images for microrobot A and microrobot B are shown in Fig. 5e, f. During the test, three groups of tests are conducted by randomly selecting 50 images to calculate the average error of pose estimation. The mean errors and standard deviation for the quantitative evaluation of microrobot pose estimation based on the proposed method and the traditional template matching method are shown in Table 1.

The results indicate that the average pose estimation errors for microrobot A and B are 3.31° and 3.23° respectively when using the proposed method. As for the template matching approach, the average pose estimation errors are higher compared to the ones obtained via our proposed method (5.13° vs. 3.23° for microrobot A; 6.43° vs. 3.50° for microrobot B).

Moreover, the computation time for pose estimation of the microrobots based on the proposed method and the template matching method is 0.002 and 0.028 seconds respectively. The computation time of template matching is much longer compared to the proposed method, which limits its online applications with real-time computation requirements. Therefore, the proposed sim-to-real learning-to-match approach can yield higher accuracy and require less computation time, which is essential for online pose estimation of micro/nano-objects.

Results for OM images. For the data collected via the OM, we define two cases for experimental validation, which depend on how we organize the training dataset and the testing dataset.

- Case A: for the training dataset, k is set to $10°$ and K is 5. According to Eq. (1), 10 different classes of microrobots with different pose values are included in the training dataset, while 50 microscopic images of a microrobot were collected
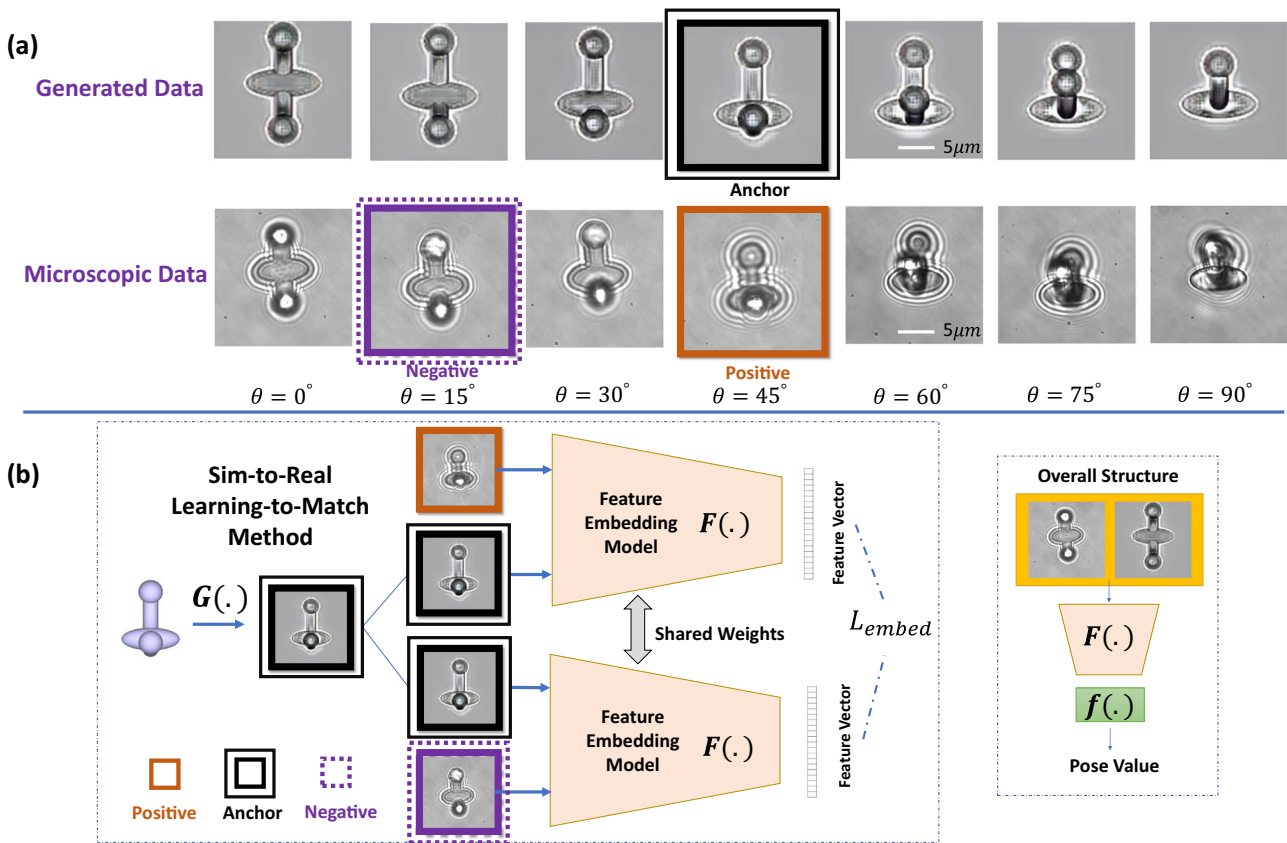
Fig. 3 Overview of the learning-to-match model. a Examples of the generated data obtained via sim-to-real transfer based on CycleGAN and the corresponding experimental data with the same pose. An example of the sampling of an anchor frame with $\theta = 45°$, a positive frame with $\theta = 45°$ and a negative frame with $\theta' = 15°$. **b** Concept illustration of the learning-to-match approach and overall structure of the pose estimation model. $G(.)$ represents the generator; $F(.)$ is the feature embedding model; $f(.)$ is the multi-layer perception (MLP) network.
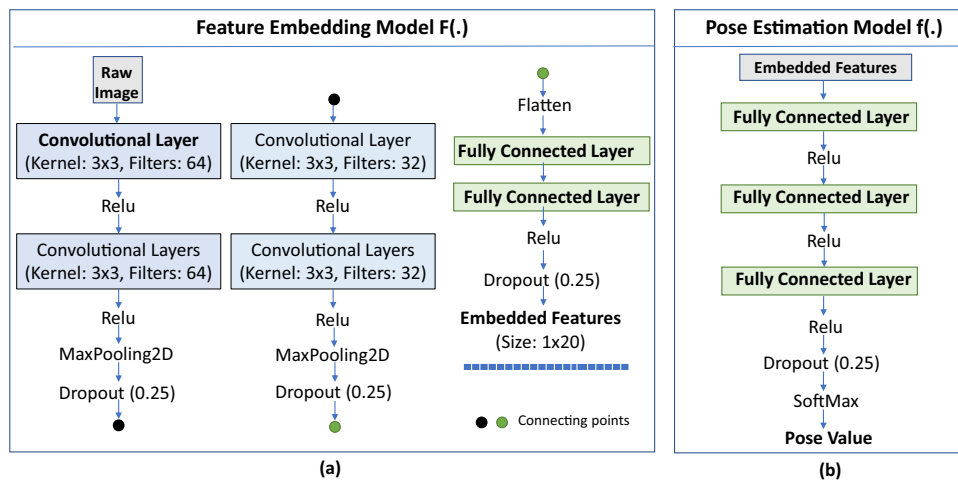

Fig. 4 Neural network architecture for the implementation of learning-to-match approach. a The feature embedding model. **b** A multi-layer perception (MLP) model for pose estimation.

in total. For the testing dataset, k is set as 5°, K is set as 8. According to Eq. (1), 19 different classes of microrobots with different pose values are included in the testing dataset. Different from data included in the training dataset, another 152 OM images of a microrobot were collected, which includes new values of pose that have not been included in the training dataset. In this case, microrobots with pose values of $\theta = 0°$, $10°$, $\cdots$, $80°$, $90°$ are used for model training.

During the testing phase, the pre-trained model can be used for microrobot pose estimation with labels of $\theta = 5°, 15°, \cdots ,$ $75°$, $85°$, the data of which are not included in the model training process.

- Case B: for both training and testing dataset, k is set to 5°, K is set as 8. According to Eq. (1), 19 different classes of microrobots with different pose values are included in the testing dataset. In all, 152 microscopic images of a microrobot
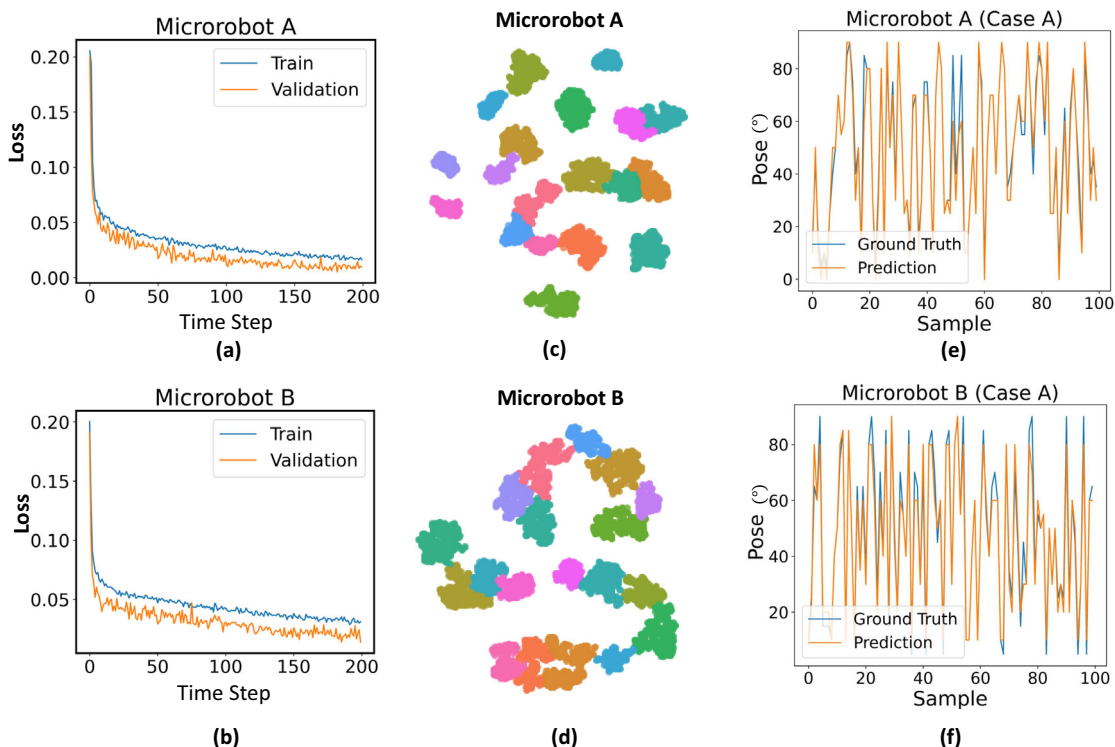
Fig. 5 Model training and results for the evaluation of the proposed method using scanning electron microscope (SEM) data. a, b The training and validation loss for learning-to-match model of microrobot A and microrobot B respectively. **c, d** Clustering results based on t-distributed stochastic neighbor embedding (t-SNE) for microrobot A and microrobot B respectively after feature extraction. Different colors represent different clusters with different pose values. The qualitative results indicate that the feature embedding model is desired to separate the features of microrobots with different poses to different clusters. **e, f** Comparisons between ground truth data and predicted results of pose values for microrobot A, microrobot B respectively.

Table 1 Mean errors for pose estimation of microrobots using scanning electron microscopy (SEM) data.

Average errors	Microrobot A	Microrobot B
Proposed method	3.23°	3.50°
Template matching	5.13°	6.43°

were collected in total. After constructing the dataset with data augmentation, 80% of the data was used to construct the training dataset while the remaining 20% was used to construct the testing dataset.

For experiments in both cases, the images of a microrobot have either different values of pose or depth levels, which means that the training and testing data have significant differences. For Case A, the number of classes used for training and testing is different. The model is trained by using the data obtained from 10 different classes of pose values, while tested by using the data obtained from 19 different classes of pose values. This demonstrates the advantages of the proposed method. During the model training process, we do not need to collect data with all the labels that we need to predict during the testing phase. The training process is target at finding a reasonable feature embedding method to map the data to a feature vector that contains useful information for few-shot adaptation using the MLP model for pose prediction. During the model training, the generated data contains pose values of $\theta = 0°, 5°, 10°, 15°, \cdots , 75°, 80°, 85°, 90°$, while the real data (experimental data) of microrobots contains pose values of

$\theta = 0°, 10°, \cdots , 80°, 90°$. During the testing phase, when given new experimental data of microrobot with new pose values of $\theta = 5°, 15°, \cdots , 75°, 85°$ that have not been demonstrated during the model training process, the embedding model can map this new data to the generated data with the same pose labels. Following that, the extracted feature vectors can lead to high accuracy of pose prediction, even though the data with new labels have not been demonstrated during the model training process.

As for Case B, the number of classes used for training and testing is the same. Experiments of Case A are designed to demonstrate the data-efficiency of the proposed method, which indicates that we do not need to collect the images of every specific pose of the microrobots to construct the training dataset, while the proposed method can still be effective for pose estimation when unseen pose values present in the testing dataset.

The training and validation loss of embedding model training using OM data for microrobot A and B in both cases are shown in Supplementary Fig. 1. For microrobot A and B in Case A, the training loss is reduced from the original value of 0.12 to 0.02. As for the validation, the loss value is reduced from 0.08 and converged to 0.01. This means that the embedding model is effective to capture similar features from both the experimental data and the generated data, and can discover the differences between microrobots with different pose values across both domains. The same conclusions can be drawn for the model training in Case B using the dataset of microrobot A and B.

The clustering results based on t-SNE dimension reduction is shown in Fig. 6a, which indicates that the feature embedding model can map the microrobots with the same pose to the same cluster, while the distance between different centers of the clusters
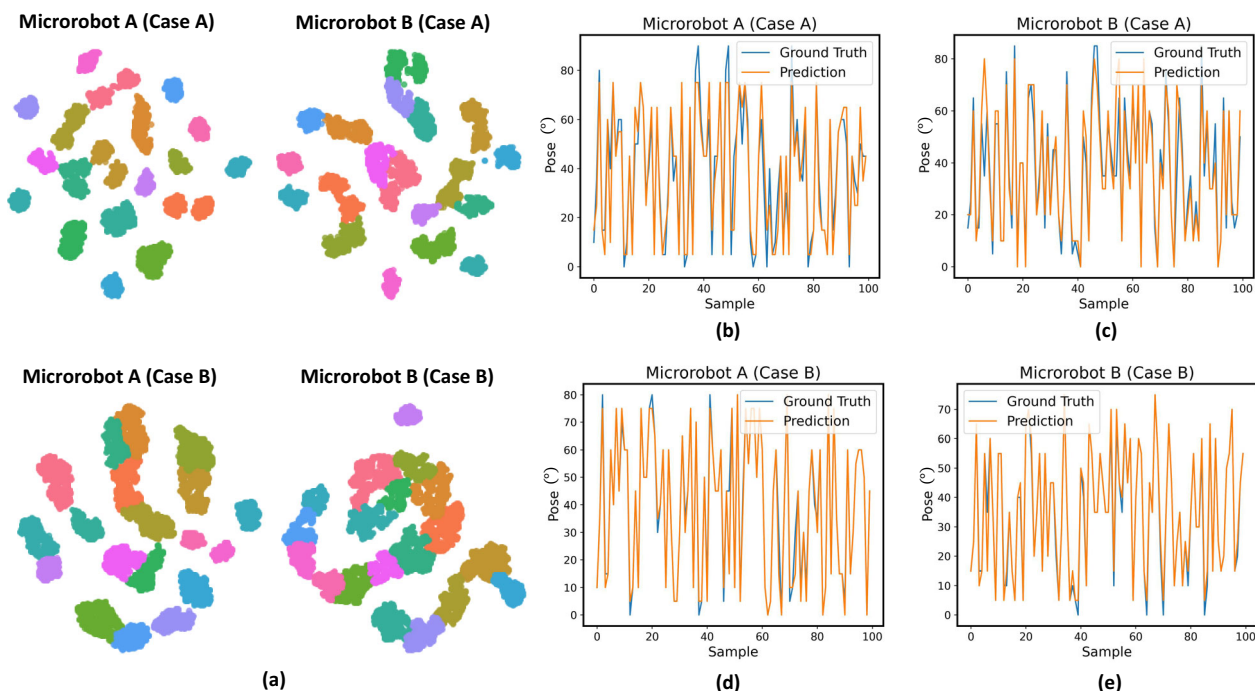
Fig. 6 Model training and results for the evaluation of the proposed method using optical microscope (OM) data. a The clustering results based on t-distributed stochastic neighbor embedding (t-SNE) dimension reduction for OM data. Different colors represent different clusters with different pose values. Comparison between ground truth data and predicted results of pose values for OM images using the testing dataset for **b** microrobot A in Case A, **c** microrobot B in Case A, **d** microrobot A in Case B, **e** microrobot B in Case B.

is evident. This means that the feature embedding model is desired to separate the features of microrobots with different poses to different clusters.

Validation is conducted using the testing dataset for comparisons between the ground truth data and predicted results for robot out-of-plane pose estimation, where 100 data points are shown in Fig. 6b–e. It can be seen that in both cases, the predicted pose values are almost similar to the ground-truth values. The quantitative evaluation results are shown in Table 2, where the average errors of pose estimation for microrobots using the OM images are calculated. Template matching based on normalized correlation matching approach is used as the baseline for comparative study[52]. S→M represents using the simulation data as the templates for pose estimation of the experimental data, while M→M represents using the experimental data as the templates.

The results indicate the applicability of the proposed method for pose estimation of microrobots under OM, since the pose estimation error is within 10°, which cannot be differentiated by the operators' eyes. In Case B, the average pose estimation error for microrobot A and microrobot B is 1.48° and 1.29° respectively. The pose estimation errors in Case A are higher than that in Case B, since there are some unseen pose values in Case A and the data for training is less than that in Case B.

Due to the domain gap between the simulation data and the experimental data, it does not make sense to use the images obtained from simulation as templates and apply them to experimental data for template matching. The pose estimation accuracy is high when using templates from the experimental data. For microrobot A and microrobot B, the average errors of pose estimation using labeled simulation data as templates are 32.39° and 30.17° respectively, while the average errors can be reduced to 6.03° and 13.81° respectively when using labeled experimental data as templates. However, it is difficult to collect all the image data with different poses and depth levels during experiments for template

Table 2 Results for pose estimation of microrobots using optical microscope (OM) data.

	Microrobot A	Microrobot B
Proposed method (Case A)	5.69°	5.16°
Proposed method (Case B)	1.48°	1.29°
Template matching (S→M)	32.39°	30.17°
Template matching (M→M)	6.03°	13.81°

matching, which can be known as one of the limitations of using this approach for the pose estimation of micro/nanorobots.

Discussion

Few-shot learning represents a type of machine learning where the training dataset contains limited labeled data for different classes, contrary to the conventional deep learning which employs a large volume of data for model training. The problem addressed in this paper is related to supervised domain adaptation for few-shot learning, where only very few target labeled data are available for model training. However, the model trained on one robot cannot be applied to another microrobot directly with different or more complex shapes without retraining. For a new robot, we need several labeled image data for calibration. With our proposed method based on few-shot learning concept, we eliminate the need of collecting a large amount of data for supervised learning.

Transfer learning method can be used as an extension of the proposed method. That is to say, the model obtained for angle prediction of microrobot A can be fine-tuned and applied to angle prediction of microrobot B, which saves time for model re-training. The accuracy for angle prediction can reach similar performance, while the computation time can be reduced. To

further enhance the generalizability of the proposed method, meta learning approach can be investigated. Meta learning, such as model agnostic meta learning[53], can be used to enable the proposed method to be adapted to the pose estimation for multiple microrobots with ease.

The quality of SEM images may get affected by the electrical charging of the samples or other environmental factors. The issues of image drift may cause inaccuracy of the pose estimation results without in situ calibration. Therefore, the robustness of the proposed methods can be further enhanced by automatic artefact removal methods. Real-time monitoring of micro/nano-robots with precise tracking and pose estimation for closed-loop control can be investigated, which is the first step towards the construction of intelligent and versatile SEM-based or OM-based micro/nano-robotic platforms for nanoscience or biomedical applications.

To summarize, we have proposed a sim-to-real learning-to-match model in this article, which enables micro/nano-object pose estimation based on limited labeled experimental data, while simulated data is used to enlarge the dataset for training. The domain gap between the simulated data and the experimental data is reduced via CycleGAN, which implements sim-to-real transfer to translate the simulated data to the experimental data with corresponding labels to form a new enlarged dataset. To further minimize the domain gap, a learning-to-match approach is developed to train a feature embedding model to map the generated data and the experimental data to the same low-dimensional space. Combining the experimental data and the generated data, the new dataset is compressed via the feature embedding model, and is employed to train a simple MLP model for micro/nano-object pose estimation. In addition, we conduct a series of ablation studies (see Supplementary Notes 1–4). The results of which are detailed in Supplementary Tables 1–4.

Two microrobots with different shapes were fabricated and used for experimental validation. Both the SEM and OM images were collected for model training. Comparisons are made between the template matching approach and the proposed approach. Results indicated that few-shot learning can be implemented for the pose estimation of microrobots using the proposed method. The pose estimation error for SEM images is smaller than 4°, which is considerably better than those using the template matching approach. The pose estimation error for OM images is within a reasonable range (<6°).

For an SEM-integrated micro/nanomanipulation system, the operator normally relies on the monocular view for the perception of the target micro/nano-objects for operation. To observe the samples from different views, the stage that is used to hold the specimen (micro/nano-objects) is required to be tilted. However, the adjustment of the tilting angle is not intuitive, and cannot be applied for real-time operation. Therefore, with the pose estimation method for micro/nano-object or robotic end-effectors, we can provide a 3D virtual views generation interface for SEM-integrated micro/nano-manipulation, through which we can observe the target object with desired customized viewing angle. The details for this application is illustrated in Supplementary Note 6 as an example, where the results are shown in Supplementary Fig. 7.

This proposed method allows pose estimation of micro/nano-objects using a single image obtained from SEM or OM as the visual feedback. This work is applicable to transmission electron microscopy (TEM) based systems or other imaging systems. Moreover, it can be extended to many applications which may involve micro/nano-robotic systems, and benefit other research fields.

Methods

Microrobot fabrication. The microrobots used for experimental verification were fabricated using the Two-Photon Polymerization[54]. Photoresist (Nanoscribe, IP-L 780) was used as the material for printing the microrobots via the micro 3D printing system (Nanoscribe GmbH, Germany). The details of the printing process can be found in our previous work[11].

Data collection. For the data collection of the microrobots with different poses via SEM, 12.0 kx magnification was used, while the high voltage (HV) was set to be 5.0 kV and working distance (WD) was set to be ~13.7 mm.

Image preprocessing. Image preprocessing with data augmentation is necessary before model training to reduce the noises in the data. Given the images collected during experiments, a bounding box is manually placed to identify the initial position of the microrobot of interest. Gaussian filter is applied to remove the noises from the images.

Subsequently, a binary segmentation of the microrobot is generated by thresholding intensity. The threshold is manually tuned to segment the main body of the microrobot. Illustration of the threshold tuning process is shown in Supplementary Fig. 3. The original threshold of intensity is set to be 210, and is gradually decreased until the segmented main body of the microrobot has a clear boundary. For example, 120 is used as the threshold for microrobot B during image preprocessing.

Subsequently, the 2D position of the centroid $[x_c, y_c]$ of the segmented microrobot can be computed from the center of mass of this binary image. Each image is cropped to have the dimension of 256×256 pixels, the central point of which is coincided with $[x_c, y_c]$. To this end, we can crop the image with the microrobot located in the central area of the image. To reduce the computation time, the cropped image is resized to 100×100 pixels. Illustration of the image preprocessing is shown in Supplementary Fig. 4, where microrobot B is used as an example.

Data augmentation is performed to enlarge the dataset via horizontal flipping, translation with a range of 20 pixels. For both the training and testing data, pre-processing of the image is conducted. Pixel intensities are rescaled to the range of $[-0.5, 0.5]$ as follows.

$$p(x', y') = p(x, y)/255 - 0.5 \qquad (4)$$

Loss functions definitions. Suppose that n is the total number of samples used for calculating the loss function, the adversarial loss on the observation samples in domain $\boldsymbol{M}$ can be calculated as follows:

$$\mathcal{L}_{adv}(\boldsymbol{G}, \boldsymbol{D_M}, \boldsymbol{S}, \boldsymbol{M}) = \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{D_M}(x_i^m) - 1\right)^2 + \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{D_M}(\boldsymbol{G}(x_i^s))\right)^2 \qquad (5)$$

Similarly, the adversarial loss on the observation samples in domain $\boldsymbol{S}$ can be calculated as follows:

$$\mathcal{L}_{adv}(\boldsymbol{G'}, \boldsymbol{D_S}, \boldsymbol{M}, \boldsymbol{S}) = \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{D_S}(\boldsymbol{G'}(x_i^m))\right)^2 + \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{D_S}(x_i^s) - 1\right)^2 \qquad (6)$$

The cycle consistency loss can be calculated as follows.

$$\mathcal{L}_{cyc}(\boldsymbol{G}, \boldsymbol{G'}) = \mathcal{L}_{cyc}^1 + \mathcal{L}_{cyc}^2 = \frac{1}{n}\sum_{i=1}^{n}\left[\|\boldsymbol{G'}(\boldsymbol{G}(x_i^s)) - x_i^s\|_1\right]$$
$$+ \frac{1}{n}\sum_{i=1}^{n}\left[\|\boldsymbol{G}(\boldsymbol{G'}(x_i^m)) - x_i^m\|_1\right] \qquad (7)$$

where $\|.\|$ represents the L1 norm (Manhattan norm). The overall loss is computed by adding the adversarial loss of $\boldsymbol{G}$ and $\boldsymbol{G'}$ as well as the cycle consistency loss, which is defined as follows:

$$\mathcal{L}(\boldsymbol{G}, \boldsymbol{G'}, \boldsymbol{D_S}, \boldsymbol{D_M}) = \mathcal{L}_{adv}(\boldsymbol{G}, \boldsymbol{D_M}, \boldsymbol{S}, \boldsymbol{M}) + \mathcal{L}_{adv}(\boldsymbol{G'}, \boldsymbol{D_S}, \boldsymbol{M}, \boldsymbol{S}) + \lambda\mathcal{L}_{cyc}(\boldsymbol{G}, \boldsymbol{G'}) \qquad (8)$$

where λ is a parameter for controlling the relative importance between the adversarial loss and the cycle consistency loss.

The loss function $\mathbb{L}$ for training $F(.)$ is listed as follows.

$$\mathbb{L} = \sum_{j}^{J} max(\|F(X_j^a) - F(X_j^p)\|_2^2 - \|F(X_j^a) - F(X_j^n)\|_2^2 + \phi, 0) \qquad (9)$$

Model training. The model was implemented in Python based on Keras[55], and was trained on a PC with an Intel Core i5-8300H CPU (2.30 GHz), a GeForce GTX 1050 GPU (NVidia Corporation) and 8 GB of RAM.

The model was trained for 200 epochs with a learning rate of 0.0001 based on the Adam optimizer, while the batch size was set to be 80. The loss function was constructed by mean-square-error (MSE) for feature embedding model training.

Data availability

The data that support the findings of this study are available from the corresponding author upon request.

Code availability

The code that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. Nakajima, M., Arai, F., Dong, L., Nagai, M. & Fukuda, T. Hybrid nanorobotic manipulation system inside scanning electron microscope and transmission electron microscope. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, vol. 1, 589–594 (IEEE, 2004).
2. Zimmermann, S., Tiemerding, T. & Fatikow, S. Automated robotic manipulation of individual colloidal particles using vision-based control. *IEEE/ASME Trans. Mechatron.* 20, 2031–2038 (2014).
3. Bartenwerfer, M. et al. Design of a micro-cartridge system for the robotic assembly of exchangeable afm-probe tips. In *2013 IEEE International Conference on Robotics and Automation*, 1730–1735 (IEEE, 2013).
4. Dong, L., Arai, F. & Fukuda, T. Nanoassembly of carbon nanotubes through mechanochemical nanorobotic manipulations. *Jpn J. Appl. Phys.* 42, 295 (2003).
5. Ru, C. et al. Automated four-point probe measurement of nanowires inside a scanning electron microscope. *IEEE Trans. Nanotechnol.* 10, 674–681 (2010).
6. Zhu, Y. & Espinosa, H. D. An electromechanical material testing system for in situ electron microscopy and applications. *Proc. Natl Acad. Sci. USA* 102, 14503–14508 (2005).
7. He, R. & Yang, P. Giant piezoresistance effect in silicon nanowires. *Nat. Nanotechnol.* 1, 42–46 (2006).
8. Abrahamians, J.-O., Sauvet, B., Polesel-Maris, J., Braive, R. & Régnier, S. A nanorobotic system for in situ stiffness measurements on membranes. *IEEE Trans. Robot.* 30, 119–124 (2013).
9. Mazerolle, S. et al. Nanomanipulation in a scanning electron microscope. *J. Mater. Process. Technol.* 167, 371–382 (2005).
10. Hou, J. et al. Afm-based robotic nano-hand for stable manipulation at nanoscale. *IEEE Trans. Autom. Sci. Eng.* 10, 285–295 (2012).
11. Zhang, D., Barbot, A., Lo, B. & Yang, G. Distributed force control for microrobot manipulation via planar multi spot optical tweezer. *Adv. Opt. Mater.* 8, 2000543 (2020).
12. Wang, M. et al. System calibration towards automated nanomanipulation inside scanning electron microscope. In *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, 1135–1140 (IEEE, 2017).
13. Bolopion, A., Xie, H., Haliyo, D. S. & Régnier, S. Haptic teleoperation for 3-d microassembly of spherical objects. *IEEE/ASME Trans. Mechatron.* 17, 116–127 (2010).
14. Fatikow, S. et al. Depth-detection methods for cnt manipulation and characterization in a scanning electron microscope. In *2007 International Conference on Mechatronics and Automation*, 45–50 (IEEE, 2007).
15. Shi, C. et al. Recent advances in nanorobotic manipulation inside scanning electron microscopes. *Microsyst. Nanoeng.* 2, 1–16 (2016).
16. Wang, H. et al. Automated assembly of vascular-like microtube with repetitive single-step contact manipulation. *IEEE Trans. Biomed. Eng.* 62, 2620–2628 (2015).
17. Sievers, T. & Fatikow, S. Pose estimation of mobile microrobots in a scanning electron microscope. In *Proc. Int. Conference on Informatics in Control, Automation and Robotics (ICINCO'05)*, 193–198 (2005).
18. Sievers, T. & Fatikow, S. Real-time object tracking for the robot-based nanohandling in a scanning electron microscope. *J. Micromechatronics* 3, 267–284 (2006).
19. Grammatikopoulou, M. & Yang, G.-Z. Three-dimensional pose estimation of optically transparent microrobots. *IEEE Robot. Autom. Lett.* 5, 72–79 (2019).
20. Zarrouk, A., Belharet, K. & Tahri, O. Vision-based magnetic actuator positioning for wireless control of microrobots. *Robot. Auton. Syst.* 124, 103366 (2020).
21. Palagi, S., Jager, E. W., Mazzolai, B. & Beccai, L. Propulsion of swimming microrobots inspired by metachronal waves in ciliates: from biology to material specifications. *Bioinspir. Biomim.* 8, 046004 (2013).
22. Kudryavtsev, A. V., Dembele, S. & Piat, N. Full 3d rotation estimation in scanning electron microscope. In *IEEE/RSJ International Conference on Intelligent Robots and Systems* (2017).
23. Bergeles, C., Kratochvil, B. E. & Nelson, B. J. Visually servoing magnetic intraocular microdevices. *IEEE Trans. Robot.* 28, 798–809 (2012).
24. Rivenson, Y. et al. Deep learning enhanced mobile-phone microscopy. *ACS Photon.* 5, 2354–2364 (2018).
25. Wu, Y. et al. Deep learning enables high-throughput analysis of particle-aggregation-based biosensors imaged using holography. *ACS Photon.* 6, 294–301 (2018).
26. Wu, Y. et al. Label-free bioaerosol sensing using mobile microscopy and deep learning. *ACS Photon.* 5, 4617–4627 (2018).
27. Fu, T., Zang, Y., Zou, Q., Nuckolls, C. & Venkataraman, L. Using deep learning to identify molecular junction characteristics. *Nano Lett.* 20, 3320–3325 (2020).
28. Cao, B. et al. How to optimize materials and devices via design of experiments and machine learning: Demonstration using organic photovoltaics. *ACS Nano* 12, 7434–7444 (2018).
29. Brown, K. A., Brittman, S., Maccaferri, N., Jariwala, D. & Celano, U. Machine learning in nanoscience: Big data at small scales. *Nano Lett.* 20, 2–10 (2019).
30. Rashidi, M. & Wolkow, R. A. Autonomous scanning probe microscopy in situ tip conditioning through machine learning. *ACS Nano* 12, 5185–5189 (2018).
31. Voznyy, O. et al. Machine learning accelerates discovery of optimal colloidal quantum dot synthesis. *ACS Nano* 13, 11122–11128 (2019).
32. Lee, C.-H. et al. Deep learning enabled strain mapping of single-atom defects in two-dimensional transition metal dichalcogenides with sub-picometer precision. *Nano Lett.* 20, 3369–3377 (2020).
33. Xiang, Y., Schmidt, T., Narayanan, V. & Fox, D. Posecnn: a convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199* (2017).
34. Kehl, W., Manhardt, F., Tombari, F., Ilic, S. & Navab, N. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *Proceedings of the IEEE international conference on computer vision*, 1521-1529 (2017).
35. Rad, M. & Lepetit, V. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *Proceedings of the IEEE International Conference on Computer Vision*, 3828-3836 (2017).
36. Pavlakos, G., Zhou, X., Chan, A., Derpanis, K. G. & Daniilidis, K. 6-dof object pose from semantic keypoints. In *2017 IEEE international conference on robotics and automation (ICRA)*, 2011-2018 (IEEE, 2017).
37. Fei-Fei, L., Fergus, R. & Perona, P. One-shot learning of object categories. *IEEE Trans. pattern Anal. Mach. Intell.* 28, 594–611 (2006).
38. Hinterstoisser, S., Lepetit, V., Wohlhart, P. & Konolige, K. On pre-trained image features and synthetic images for deep learning. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 0-0 (2018).
39. Patel, V. M., Gopalan, R., Li, R. & Chellappa, R. Visual domain adaptation: A survey of recent advances. *IEEE signal Process. Mag.* 32, 53–69 (2015).
40. Volpi, R., Morerio, P., Savarese, S. & Murino, V. Adversarial feature augmentation for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5495-5504 (2018).
41. Gong, B., Shi, Y., Sha, F. & Grauman, K. Geodesic flow kernel for unsupervised domain adaptation. In *2012 IEEE conference on computer vision and pattern recognition*, 2066-2073 (IEEE, 2012).
42. Zhu, J.-Y., Park, T., Isola, P. & Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, 2223-2232 (2017).
43. Bousmalis, K. et al. Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In *2018 IEEE international conference on robotics and automation (ICRA)*, 4243-4250 (IEEE, 2018).
44. Zhang, D., Lo, P. W., Zheng, J. Q., Bai, W. & Lo, B. Data-driven microscopic pose and depth estimation for optical microrobot manipulation. *Acs Photonics* 7, 3003–3014 (2020).
45. Guo, X., Wang, Z., Yang, Q., Lv, W. & Huang, J. Gan-based virtual-to-real image translation for urban scene semantic segmentation. *Neurocomputing* 394 (2019).
46. Isola, P., Zhu, J.-Y., Zhou, T. & Efros, A. A. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1125-1134 (2017).
47. Zhu, J.-Y., Park, T., Isola, P. & Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, 2223-2232 (2017).
48. Kim, T., Cha, M., Kim, H., Lee, J. K. & Kim, J. Learning to discover cross-domain relations with generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 1857-1865 (JMLR. org, 2017).
49. Yi, Z., Zhang, H., Tan, P. & Gong, M. Dualgan: Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE international conference on computer vision*, 2849-2857 (2017).
50. Van der Maaten, L. & Hinton, G. Visualizing data using t-sne.Journal of machine learning research 9 (2008).
51. Van Der Maaten, L. Accelerating t-sne using tree-based algorithms. *J. Mach. Learn. Res.* 15, 3221–3245 (2014).
52. Strickland, R. N., Draelos, T. & Mao, Z. Edge detection in machine vision using a simple l1 norm template matching algorithm. *Pattern Recognit.* 23, 411–421 (1990).
53. Finn, C., Abbeel, P. & Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, 1126-1135 (PMLR, 2017).
54. Kawata, S., Sun, H.-B., Tanaka, T. & Takada, K. Finer features for functional microdevices. *Nature* 412, 697 (2001).

55. Gulli, A. & Pal, S. *Deep learning with Keras* (Packt Publishing Ltd, 2017).

Author contributions
D.Z. proposed the method, developed the algorithm, implemented the algorithm for experimental validation. A.B., W.B., F.P.W.L., G.-Z.Y., and B. L. provided valuable comments on the manuscript. F.S. fabricated the experimental samples and collected the SEM images. Correspondence and requests for materials should be addressed to D.Z.

Competing interests
The authors declare no competing interests.

Additional information
Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42005-022-00844-z.

Correspondence and requests for materials should be addressed to Dandan Zhang or Guang-Zhong Yang.

Peer review information *Communications Physics* thanks Alexander Krull, Giovanni Volpe and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.