Self-Supervised Rigid Registration for Small Images

Ruoxin Ma[1], Shengjie Zhao[1,2*], and Samuel Cheng[3]
[1] School of Electronics and Information Engineering, Tongji University
Shanghai, 201804, China
[e-mail: 1930811@tongji.edu.cn]
[2] School of Software Engineering, Tongji University
Shanghai, 201804, China
[e-mail: shengjiezhao@tongji.edu.cn]
[3] School of Electrical and Computer Engineering, University of Oklahoma
Tulsa, OK 74135, USA
[e-mail: samuel.cheng@ou.edu]
*Corresponding author: Shengjie Zhao

Received November 26, 2020; revised December 17, 2020; accepted December 22, 2020; published January 31, 2021

Abstract

For small image registration, feature-based approaches are likely to fail as feature detectors cannot detect enough feature points from low-resolution images. The classic FFT approach's prediction accuracy is high, but the registration time can be relatively long, about several seconds to register one image pair. To achieve real-time and high-precision rigid registration for small images, we apply deep neural networks for supervised rigid transformation prediction, which directly predicts the transformation parameters. We train deep registration models with rigidly transformed CIFAR-10 images and STL-10 images, and evaluate the generalization ability of deep registration models with transformed CIFAR-10 images, STL-10 images, and randomly generated images. Experimental results show that the deep registration models we propose can achieve comparable accuracy to the classic FFT approach for small CIFAR-10 images (32×32) and our LSTM registration model takes less than 1ms to register one pair of images. For moderate size STL-10 images (96×96), FFT significantly outperforms deep registration models in terms of accuracy but is also considerably slower. Our results suggest that deep registration models have competitive advantages over conventional approaches, at least for small images.

Keywords: Rigid Registration, Self-Supervised Learning, Small Image, LSTM, Homography Estimation

This work was supported in part by the National Key Research and Development Project under Grant 2019YFB2102300, 2019YFB2102301, in part by the National Natural Science Foundation of China under Grants 61936014, in part by the Fundamental Research Funds for the Central Universities.

http://doi.org/10.3837/tiis.2021.01.011 **ISSN : 1976-7277**

1. Introduction

Image registration [1] is a fundamental computer vision [2, 3] task and is widely used in remote sensing [4, 5], 3D reconstruction [6], SLAM [7, 8] and medical image processing [9, 10]. For example, registered image pairs across different times or subjects can be used for diagnostic guidance or minimally invasive surgery [11, 12].

Spatial transformations can be divided into rigid transformation, similarity transformation, affine transformation, and projection transformation based on the degree of freedom of the homography matrix. Furthermore, image registration is to obtain the spatial transformation which aligns the image pairs best.

Feature-based registration approaches are commonly used and robust to noise and distortion, such as SIFT [13], SURF [14], ORB [15], and AKAZE [16] algorithms, which can obtain homography matrix through feature point detection [17], feature description [18], feature matching [19], and image warping procedure. However, these approaches fail to register small images with a high probability due to the insufficient number of feature points detected for further estimation. Classic FFT [20-22] registration approach can achieve high registration accuracy with the normalized cross-correlation [23] metrics. However, it takes several seconds for the classic FFT approach to register one image pair. The FFT approach's registration accuracy decreases if the images' resolution or the number of high-frequency components decreases.

In this paper, we focus on real-time rigid registration for small images, where deep regression networks predict the registration parameters in a single forward propagation pass based on the whole image. To address the lack of datasets with ground truth, we adopt the self-supervised learning principle [24] and train deep regression networks with rigidly transformed CIFAR-10 [25] images and STL-10 [26] images. We evaluate different network architectures' generalization ability with transformed test images of the CIFAR-10 and STL-10 dataset. Moreover, we also generate random patches to access different approaches. We propose deep regression networks to achieve comparable accuracy to the classic FFT approach for small images while deep networks' registration time decreases significantly.

1.1 Related Work

With the development of deep learning, especially in the past five years, many learning-based approaches [27] have been introduced to the registration task.

Nguyen et al. [28] proposed an unsupervised homography estimation model for a robotic system. They used the 4-point parameterization and developed a layer to obtain the 3×3 homography matrix. They trained a convolutional neural network (CNN) using an L1 pixel-wise photometric loss function, different from the parameter-based loss function we used. Similarly, Rocco et al. [29] proposed a CNN for affine, homography, and thin-plate spline transformation [30]. They used the VGG-16 [31] network for feature extraction and defined the loss function based on the discrepancy between the corresponding grid points. They consider their approach as a particular case of self-supervised learning [24]. Chen et al. [32] proposed a multimodal image representation model MIRnet for slice-to-volume registration based on the self-learning strategy. They defined a specific loss function to make the paired outputs similar and retain the originals' edge information. The MIRnet is optimized in an unsupervised manner.

Despite the lack of datasets with ground truth, several researchers managed to train deep regression networks to predict the rigid registration parameters directly. Miao et al. [33] proposed a CNN regression approach to register a 3D CT or CBCT image with a 2D X-ray image. They used a patch-based registration method, different from the strategy based on the whole image we adopt. Sloan et al. [34] trained a fully convolutional neural network to register 2D to 2D images of the size 256×256. However, they allowed translation pixels and rotation angles with limited ranges of ±30 pixels and ±15 degrees. Salehi et al. [35] proposed a CNN-based registration model consisting of feature extractors and regression heads for 2D or 3D to 3D registration, predicting the translation and rotation parameters generated randomly. They compare geodesic loss with mean square error loss for training CNN architectures. At the same time, we use mean absolute error for translation pixels and rotation angles separately for 2D to 2D small image registration.

Similar to the above works, our registration models predict the rigid transformation parameters directly. However, the above works were proposed for medical image registration where the images are typically large, while our interest is to register small images such as CIFAR-10 images with 32×32 resolution. Moreover, the above works are mainly based on CNNs, while we introduce LSTM [36] architecture to 2D rigid registration task for straightforward prediction or used as a regression head.

1.2 Our Contributions

We propose different CNN and LSTM network architectures for small image rigid registration. We adopt the self-supervised learning principle and generate synthetic data by randomly translating and rotating images from the CIFAR-10 [25] and STL-10 [26] datasets.

We test our regression networks with 32×32 resolution images, including CIFAR-10 images and random patches, and 96×96 resolution images, including STL-10 images and random patches. For all these images, feature-based approaches fail to register image pairs with very high probability, and the prediction is inaccurate in rarely successful cases. Compared with the classic FFT approach, which has high accuracy and very long registration time, our deep registration architectures reduce the average registration time up to 42 folds and achieve comparable registration accuracy for smaller images (32×32) even though the FFT approach is still significantly more accurate for large images (96×96). This work suggests a novel and better way to conduct efficient rigid registration for small image patches.

2. Methods

2.1 Problem Setup

A rigid 2D transformation with translation and rotation can be represented by a homography matrix with three degrees of freedom, as below:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & \mathrm{t_y} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{1}$$

where (x, y) and (x', y') are the coordinates of the same point of the image in different coordinate systems, θ denotes the rotation angle of rigid transformation, and $(t_x, \mathrm{t_y})$ denotes the amount of translation in (x, y) direction. The goal of this work is to estimate θ, t_x and t_y from a pair of input images.

2.2 Registration Based on LSTM

We flatten the image pairs to be registered into two one-dimensional vectors, the size of which is equal to the input size of LSTM. Moreover, we treat the two vectors as two time series of LSTM. A fully connected layer is connected behind the LSTM architecture as the regression head, which has three outputs for three rigid registration parameters. **Fig. 1** shows the schematic diagram of registration by the LSTM model.

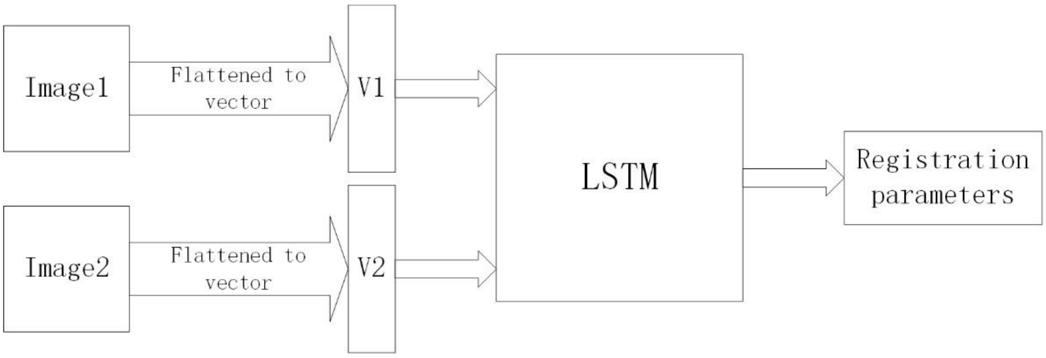

Fig. 1. Schematic diagram of registration by the LSTM model

2.3 Registration Based on CNN Extractors

We use CNN architectures with 3-channel input as extractors. The two images to be registered are fed into the same CNN extractors, obtaining two one-dimensional vectors. A fully connected layer and an LSTM architecture are adopted as the regression head. For a fully connected layer regression head, the two vectors obtained from CNN extractors will be connected into a one-dimensional vector, later fed into the regression head. For the LSTM regression head, we treat the two vectors obtained from CNN extractors as two different time series. Moreover, one fully connected layer behind the LSTM has three outputs for three rigid registration parameters. **Fig. 2** shows the schematic diagram of registration by the CNN extractor model.

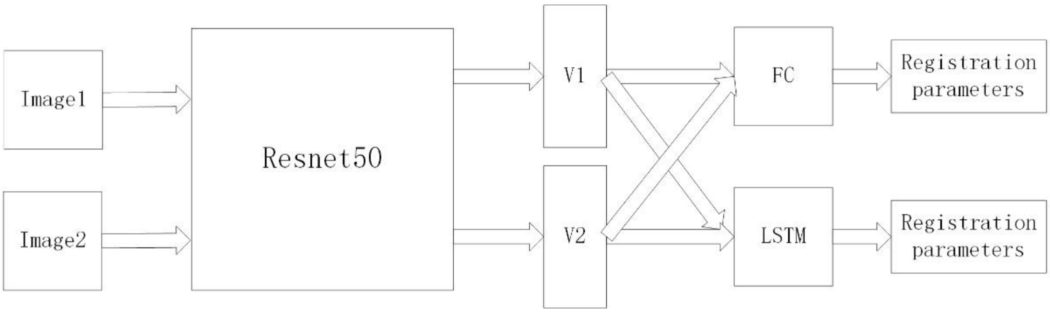

Fig. 2. Schematic diagram of registration by the CNN extractor model

2.4 Loss Function

To evaluate the registration model's performance in both translation and rotation aspects, we calculate the errors of the amount of translation and the angle of rotation separately. For translation error, we construct the ground truth vector of translation $v_{gtt} = \left(t_x, t_y\right)$ and the prediction result vector of translation $v_{pt} = (p_1, p_2)$. The test loss of translation is the mean absolute error between the first two predictions of the neural network and the ground truth vector we constructed. We calculate the test loss of translation with the formula as follows:

$$L_{translation} = \frac{1}{n}\sum_{i=1}^{n} \left|t_{x_i} - p_{1_i}\right| + \left|t_{y_i} - p_{2_i}\right| \tag{2}$$

where n is the number of the test samples, t_x, t_y denotes the amount of translation in x and y direction, p_1, p_2 represents the first and second dimension of the neural network's predictions, and the subscript i indicates the corresponding attributes of the i-th sample.

Similarly, we construct the ground truth vector of rotation $v_{gtr} = (\theta)$ and the prediction result vector of rotation $v_{pr} = (p_3)$. We calculate the test loss of rotation as follows:

$$L_{MAE_i} \equiv \left|\theta_i - p_{3_i}\right| \, mod \; 360 \tag{3}$$

$$L_{rotation} = \frac{1}{n}\sum_{i=1}^{n} L_{MAE_i} I\left(L_{MAE_i} \leq 180\right) + \left(360 - L_{MAE_i}\right) I\left(L_{MAE_i} > 180\right) \tag{4}$$

where n is the number of the test samples, θ denotes the angle of rotation, p_3 represents the third dimension of the predictions of the neural network, $I(x)$ is an indicator function, and the subscript i indicates the corresponding attributes of the i-th sample. The overall training loss is given by

$$L_{training} = \frac{1}{n}\sum_{i=1}^{n} \lambda L_{translation} + L_{rotation} \tag{5}$$

where n is the number of training samples, and λ is a hyperparameter trading-off rotation loss with translation loss.

3. Experiments

We conduct all our deep learning experiments using the PyTorch [37] framework, with the cuDNN acceleration. The GPU we use is the Nvidia Geforce GTX TITAN X. The batch size is one, the hyperparameter $\boldsymbol{\lambda}$ is set to 17, and we use Adam [38] to optimize throughout the training procedure.

3.1 Datasets

The CIFAR-10 dataset [25] consists of 50000 training samples and 10,000 test samples. The resolution of CIFAR-10 images is 32×32. The STL-10 [26] dataset has 5,000 training samples, 8,000 test samples, and 100,000 unlabeled samples, the resolution of which is 96×96. Moreover, we also generate random patches with the same resolution as the CIFAR-10 image and STL-10 image separately.

We adopt the self-supervised learning [24] principle and generate synthetic data by randomly translating and rotating the images mentioned above. Qualitative results of the LSTM registration model on the CIFAR-10 dataset are shown in **Fig. 3**.

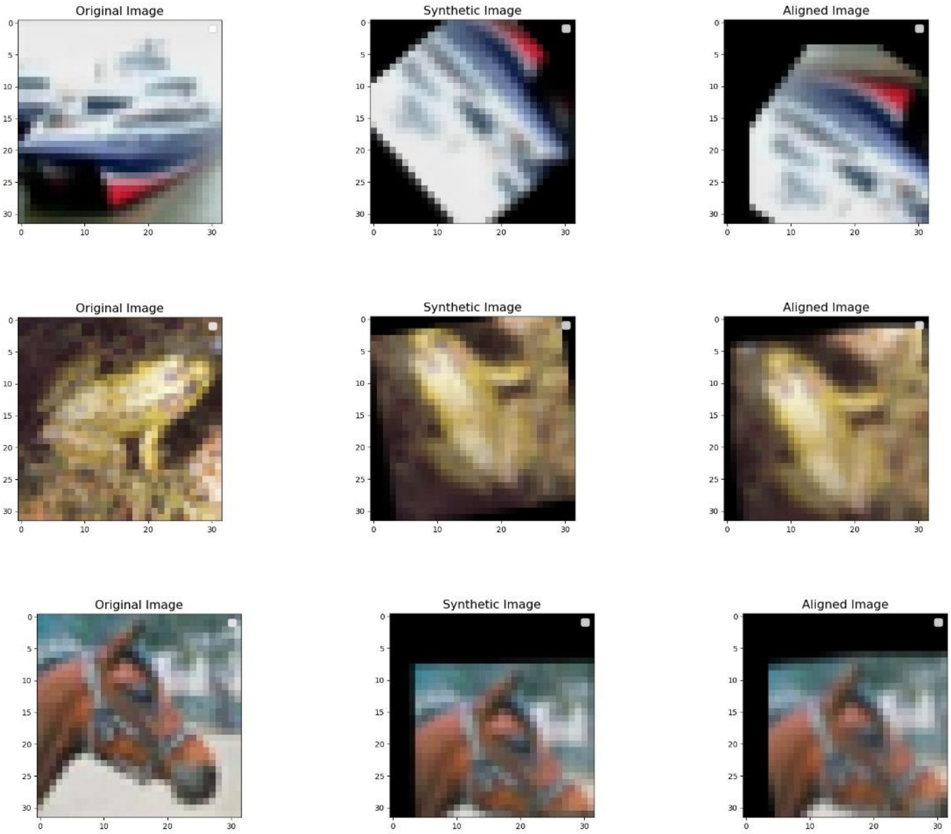

Fig. 3. Qualitative results of the LSTM registration model on the CIFAR-10 dataset

Each row shows one test sample from the CIFAR-10 dataset. The left column shows the original image, the middle column shows the synthetic image we generate by randomly translating and rotating the original image, and the right column shows the aligned image generated from the original image with the estimation parameters obtained by the LSTM registration model.

3.2 Network Architectures

3.2.1 LSTM Registration Architecture

The proposed LSTM registration model has an input layer, three LSTM hidden layers, and one fully connected layer at the end. The three LSTM hidden layers are used for feature extraction, and the last fully connected layer is used as the regression head.

The LSTM registration model's input size is 3,072, equal to the vectorized size of images of the CIFAR-10 dataset. For the small images of 32×32 resolution, the original image and the synthetic image are vectorized and fed into the input layer in two steps. Moreover, the 96×96 resolution images are down-sampled into 32×32 resolution before vectorization. The middle three layers have 3,000 LSTM hidden units in each layer. The three LSTM hidden layers provide 3,000 features for the regression head. And the last fully connected one layer has three units as outputs (corresponding to the rigid transforms' three degrees of freedom). We take the first as the translation pixels in the x-direction, the second as the translation pixels in the y-direction, and the third as the rotation angle.

3.2.2 CNN-Extractor Registration Architectures

The CNN-extractor registration network comprises the ResNeXt-50_32x4d [39] architecture as the feature extractor and the regression head, which is either one fully connected layer or the LSTM registration network we proposed above. All these architectures have been pretrained on the ImageNet [40] dataset.

We replace the final layer of the ResNeXt-50_32x4d network with a 3,072-output fully connected layer. The original image and the synthetic image are fed into the feature extractor sharing the weights separately, each for 3,072 features. The 32×32 resolution images are up-sampled into 64×64 resolution before fed into the feature extractor. We concatenate the 2×3072 features into a one-dimensional vector of size 6,144 before fed into a 3-output layer for the fully connected regression head, as described in Section 2. Moreover, the LSTM regression head has the same architecture as the LSTM registration architecture we proposed above. For the LSTM regression head, we treat the 2×3072 features as two steps, which are fed into three LSTM hidden layers for 3,000 new features. The 3,000 features are fed into the last fully connected layer which has three outputs as the estimation results.

3.3 Comparison with FFT and Feature-based Approaches

To compare the prediction accuracy and registration time of deep registration models, we implement the classic FFT approach [20, 22] and feature-based approaches. Since the FFT approach can only register images with pure translation, we rotate one image for each degree and register the other. The feature detectors we use include SIFT [13], SURF [14], ORB [15], and AKAZE [16] algorithms. We use the k-nearest neighbor [41] to match the detected keypoints and use RANSAC [42] algorithm to exclude outliers and obtain the homography matrix.

4. Experimental Results and Discussion

4.1 Evaluation of Small Images

We conduct experiments to train the deep regression networks with rigidly transformed CIFAR-10 images and test different approaches with rigidly transformed random patches of 32×32 resolution and CIFAR-10 test images under more detailed translation and rotation conditions. The prediction errors and registration times of different approaches for CIFAR-10 images and random patches are shown in **Table 1** and **Table 2**, respectively. The bracket values next to the feature-based methods indicate the rate of failure where insufficient keypoints are detected. Methods that ultimately fail to register the input are filled with N.A. The results are mean absolute values and a standard deviation of prediction errors. The translation error is measured in pixels, the rotation error is measured in degrees, and the registration time is measured in seconds.

Table 1. The prediction errors and registration time for transformed CIFAR-10 images

Model	Translation（pixels）	Rotation（degrees）	Time（s）
AKAZE (1)	N.A.	N.A.	N.A.
ORB (1)	N.A.	N.A.	N.A.
SURF (1)	N.A.	N.A.	N.A.
SIFT (0.9505)	13.3594±6.2321	62.0331±52.7100	0.0093±0.0064
FFT	1.4542±0.6620	7.9713±5.1973	2.4132±0.0896
ResNeXt-50+FC	1.0182±0.8282	6.6140±7.3644	0.0536±0.0090
ResNeXt-50+LSTM	0.8328±0.7147	5.6407±6.1125	0.0546±0.0083
LSTM	0.6641±0.4848	7.8624±8.1080	0.0007±0.0003

Table 2. The prediction errors and registration time for transformed random 32 × 32 images

Model	Translation（pixels）	Rotation（degrees）	Time（s）
AKAZE (1)	N.A.	N.A.	N.A.
ORB (1)	N.A.	N.A.	N.A.
SURF (1)	N.A.	N.A.	N.A.
SIFT (1)	N.A.	N.A.	N.A.
FFT	0.5004±0.0649	0.3596±0.1117	2.2415±0.1396
ResNeXt-50+FC	0.9879±0.9742	6.3044±7.7589	0.0502±0.0080
ResNeXt-50+LSTM	0.8320±1.0346	5.6148±7.1531	0.0528±0.0084
LSTM	1.0474±1.5365	12.1091±17.8267	0.0007±0.0003

The detailed registration results for transformed CIFAR-10 images of different registration models are shown in **Fig. 4-7**.

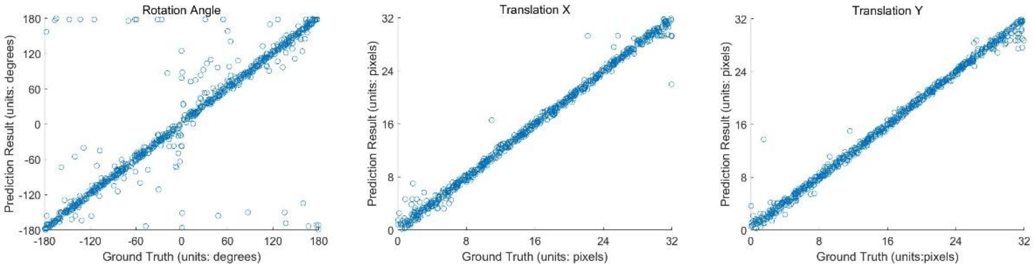

Fig. 4. The scatter plots of predicted results vs. ground truth of transformation parameters by LSTM registration model for transformed CIFAR-10 images

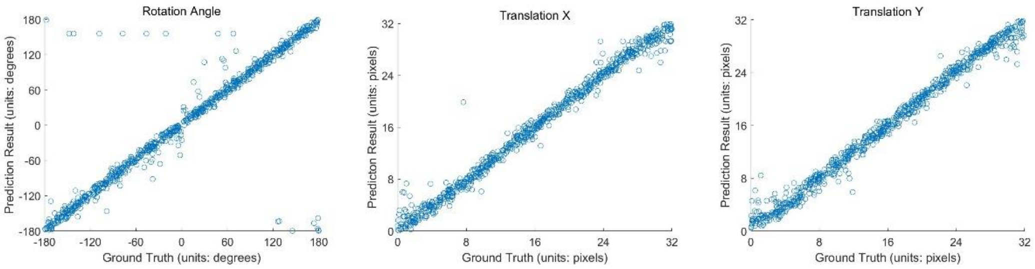

Fig. 5. The scatter plots of predicted results vs. ground truth of transformation parameters by ResNeXt-50 extractor with LSTM regression head architecture for transformed CIFAR-10 images

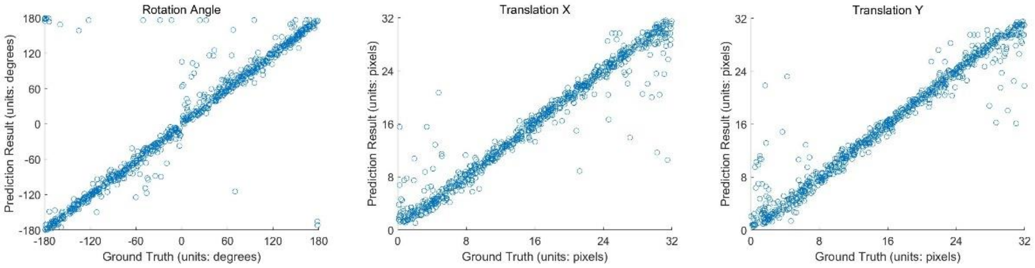

Fig. 6. The scatter plots of predicted results vs. ground truth of transformation parameters by ResNeXt-50 extractor with FC regression head architecture for transformed CIFAR-10 images

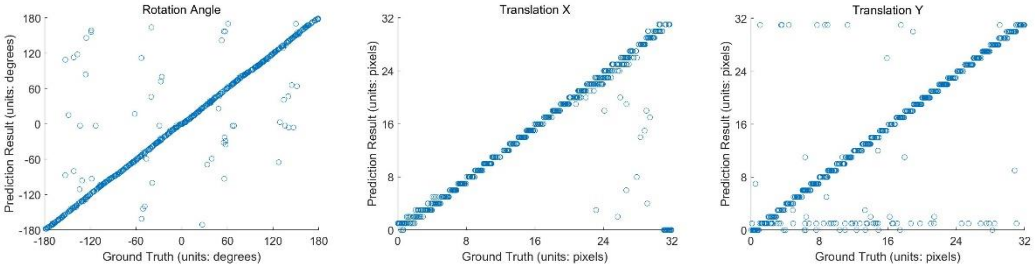

Fig. 7. The scatter plots of predicted results vs. ground truth of transformation parameters by classic FFT approach for transformed CIFAR-10 images

Fig. 4-7 shows the detailed registration results of CIFAR-10 images for translation pixels in x,y-directions and rotation angles. The errors are measured based on deviation. The registration results are more accurate if the points fit the line of y=x better. Note that the rotation error is calculated with the complementation. For example, if the ground truth of rotation angle is -180° and the prediction result is 180° , the rotation error will be 0° . The FFT's prediction results are discrete values. For the rotation estimation, the ResNeXt-50 extractor with the LSTM regression head network performs best. Moreover, the LSTM registration model performs best for the translation estimation.

We test how many keypoints feature-based approaches can detect from small images; the results are shown in **Table 3**. The results are mean absolute values and a standard deviation of keypoint numbers.

Table 3. The number of keypoints that feature-based approaches detected

Model	CIFAR-10 images	Random 32×32 images
AKAZE	0±0	0±0
ORB	0±0	0±0
SURF	0.0116±0.1071	0.0023±0.0481
SIFT	0.9010±1.5595	0.0764±0.2988

AKAZE, ORB, and SURF cannot detect sufficient feature points from all 32×32 resolution test images. SIFT only detects sufficient feature points under a few cases for the CIFAR-10 test images. For rare successful cases, the prediction errors of SIFT are unacceptably large.

The classic FFT approach's prediction accuracy is pretty good, especially for random patches of 32×32 resolution, which have more high-frequency components. It takes more than 2 seconds for the classic FFT approach to register one image pair. However, the classic FFT approach's registration accuracy is more robust than deep models under different evaluation conditions.

The proposed deep network architectures can achieve comparable accuracy to the classic FFT approach for CIFAR-10 images. The ResNeXt-50 extractor model's registration time is about 54ms, 2.24% of that of the FFT approach. Moreover, it takes 0.7ms for the LSTM registration model to register one image pair, which is 0.29% of the classic FFT approach. The deep neural networks perform similarly under different translation and rotation conditions. If the translation pixels get too large, where the registration becomes impossible, the prediction error will be much higher. For random patches of 32 × 32 resolution, there is little change of ResNeXt-50 extractor model in registration accuracy and time.

4.2 Evaluation of Large Images

We train our deep regression networks with rigidly transformed STL-10 images to evaluate these approaches' registration performance for high-resolution images. Moreover, we test feature-based approaches, the classic FFT approach, and deep registration models with rigidly transformed STL-10 images and random patches of 96×96 resolution. The prediction errors and registration times of different approaches for CIFAR-10 images and random patches are shown in **Table 4** and **Table 5**, respectively. The brackets' value next to the feature-based methods indicates the rate of failure where insufficient keypoints are detected. Methods that ultimately fail to register the input are filled with N.A. The results are mean absolute values and a standard deviation of prediction errors. The translation error is measured in pixels, the rotation error is measured in degrees, and the registration time is measured in seconds.

Table 4. The prediction errors and registration time for transformed STL-10 images

Model	Translation（pixels）	Rotation（degrees）	Time（s）
AKAZE (0.9948)	49.5539±18.8692	47.8793±45.2319	0.0090±0.0024
ORB (0.9466)	40.2261±16.9984	74.2664±58.4848	0.0128±0.0412
SURF (0.8438)	31.1103±18.0714	72.2730±60.4200	0.0106±0.0043
SIFT (0.6510)	29.6221±17.5444	67.7139±62.8882	0.0175±0.0051
FFT	0.8751±1.1441	1.0525±2.7068	11.3075±1.1275
ResNeXt-50+FC	3.0326±2.6558	6.2268±7.2041	0.0491±0.0083
ResNeXt-50+LSTM	2.7797±2.7902	6.9497±5.7735	0.0539±0.0081
LSTM	1.7943±1.0922	5.9582±5.7487	0.0009±0.0003

Table 5. The prediction errors and registration time for transformed random 96 × 96 images

Model	Translation（pixels）	Rotation（degrees）	Time（s）
AKAZE (1)	N.A.	N.A.	N.A.
ORB (0.9531)	36.4939±20.3236	72.7067±60.6450	0.0119±0.0190
SURF (0.9193)	29.0270±14.4547	76.0592±56.8022	0.0153±0.0055
SIFT (0.7969)	31.3871±19.4016	56.7080±59.4889	0.0190±0.0059
FFT	0.4982±0.0618	0.2494±0.0448	11.8021±1.8278
ResNeXt-50+FC	2.4061±2.7931	5.3728±7.3070	0.0525±0.0084
ResNeXt-50+LSTM	2.3763±2.8637	6.0183±5.4018	0.0545±0.0080
LSTM	2.4931±3.6001	7.8609±11.2270	0.0009±0.0003

The detailed registration results for transformed STL-10 images of different registration models are shown in **Fig. 8-11**.

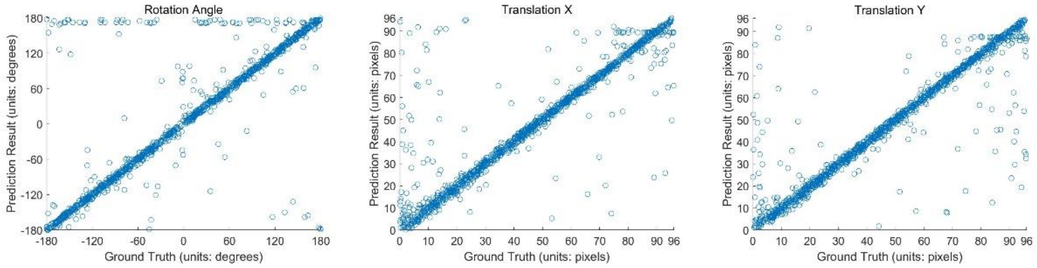

Fig. 8. The scatter plots of predicted results vs. ground truth of transformation parameters by LSTM registration model for transformed STL-10 images

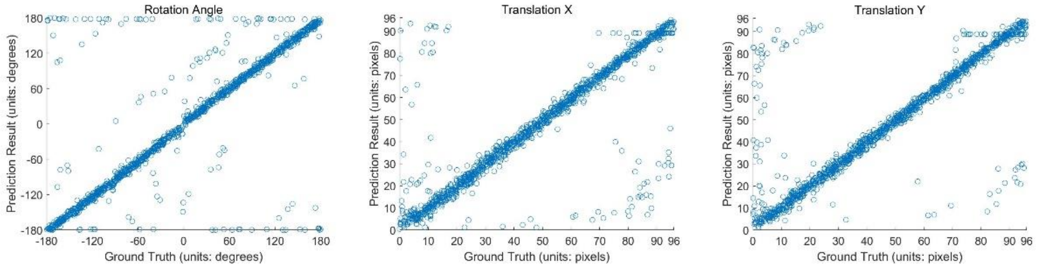

Fig. 9. The scatter plots of predicted results vs. ground truth of transformation parameters by ResNeXt-50 extractor with LSTM regression head architecture for transformed STL-10 images

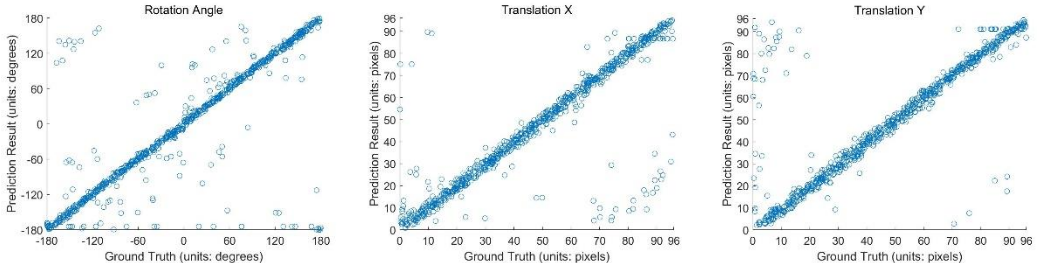

Fig. 10. The scatter plots of predicted results vs. ground truth of transformation parameters by ResNeXt-50 extractor with FC regression head architecture for transformed STL-10 images

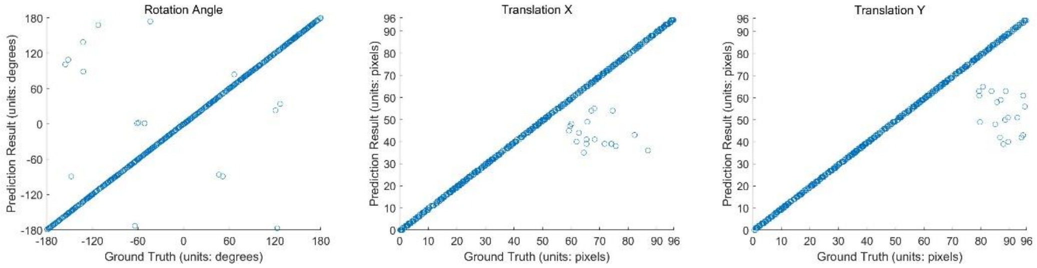

Fig. 11. The scatter plots of predicted results vs. ground truth of transformation parameters by classic FFT approach for transformed STL-10 images

Fig. 8-11 shows the detailed registration results of STL-10 images for translation pixels in x,y directions and rotation angles. The errors are measured based on deviation. The registration results are more accurate if the points fit the line of y=x better. Note that the rotation error is calculated with the complementation. For example, if the ground truth of rotation angle is -180° and the prediction result is 180°, the rotation error will be 0°. The FFT's prediction results are discrete values. The FFT approach outperforms the other approaches obviously for

STL-10 images.

We test how many keypoints feature-based approaches can detect from large images; the results are shown in **Table 6**. The results are mean absolute values and a standard deviation of keypoint numbers.

Table 6. The number of keypoints that feature-based approaches detected

Model	STL-10 images	Random 96×96 images
AKAZE	0.2083±0.6154	0.0648±0.3274
ORB	1.1181±3.1494	0.8264±2.2073
SURF	3.9792±3.6926	8.8125±20.0195
SIFT	7.8125±8.3106	3.3657±3.4143

The failure rate of feature-based approaches for 96×96 resolution images gets smaller than that for small images. However, these approaches are still likely to fail to estimate the registration parameters due to insufficient detected feature points. Moreover, the prediction errors are still substantial even with sufficient feature points detected. For these successful cases, feature-based approaches take 9-20ms to register one image pair.

The classic FFT approach achieves significantly better accuracy compared to the smaller image case and obtains the best registration accuracy among other methods. However, it takes more than 11 seconds to register one image pair.

The registration accuracy of deep regression networks decreases for 96×96 resolution images, and the registration time is almost the same as that for 32×32 resolution images. The ResNeXt-50 extractor models take about 50ms to register one image pair, and the LSTM registration model takes 0.9ms, reducing registration time up to 12,778 folds of the classic FFT approach.

5. Conclusion

In this paper, we focus on small image registration task where feature-based approaches fail due to insufficient feature points detected. We propose different network architectures for small image rigid registration and conduct experiments to train and evaluate the regression networks with images of two kinds of resolution. For 96×96 resolution images, the classic FFT approach can get the most accurate predictions, but it takes several seconds to register one image pair. Moreover, for small images of 32×32 resolution, the proposed deep regression networks can achieve comparable accuracy to the classic FFT approach with a much faster registration speed. Our experiment suggests that deep regression networks provide a competitive alternative to other approaches for small image registration, especially when image size is petite.

Future work will focus on arbitrary registration for spatial transformations with fewer constraints on the homography matrix, such as affine transformation, projection transformation, and transformations on 3D images.

References

[1] B. Zitová and J. Flusser, "Image registration methods: a survey," *Image and Vision Computing*, vol. 21, no. 11, pp. 977-1000, 2003. Article (CrossRef Link)

[2] D. A. Forsyth and J. Ponce, Computer Vision: A Modern Approach. Englewood Cliffs, NJ, USA: Prentice Hall, 2003.

[3] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, 2nd ed., Cambridge, UK: Cambridge University Press, 2006.

[4] J. Le Moigne, N. S. Netanyahu, and R. D. Eastman, Image Registration for Remote Sensing, Cambridge, UK: Cambridge University Press, 2011.

[5] J. Le Moigne, "Introduction to remote sensing image registration," in *Proc. of 2017 IEEE International Geoscience and Remote Sensing Symposium*, pp. 2565-2568, 2017. Article (CrossRef Link)

[6] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski, "Building Rome in a day," *Communications of the ACM*, vol. 54, no. 10, 2011. Article (CrossRef Link)

[7] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A Versatile and Accurate Monocular SLAM System," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147-1163, 2015. Article (CrossRef Link)

[8] J. Ma, X. Jiang, A. Fan, J. Jiang, and J. Yan, "Image Matching from Handcrafted to Deep Features: A Survey," *International Journal of Computer Vision*, 2020. Article (CrossRef Link)

[9] E. Ferrante and N. Paragios, "Slice-to-volume medical image registration: A survey," *Medical Image Analysis*, vol. 39, pp. 101-123, 2017. Article (CrossRef Link)

[10] Y. Fu, Y. Lei, T. Wang, W. J. Curran, T. Liu, and X. Yang, "Deep Learning in Medical Image Registration: A Review," *Physics in Medicine and Biology*, vol. 65, no. 20, 2020. Article (CrossRef Link)

[11] P. Markelj, D. Tomaževič, B. Likar, and F. Pernuš, "A review of 3D/2D registration methods for image-guided interventions," *Medical Image Analysis*, vol. 16, no. 3, pp. 642-661, 2012. Article (CrossRef Link)

[12] R. Liao, L. Zhang, Y. Sun, S. Miao, and C. Chefd'Hotel, "A Review of Recent Advances in Registration Techniques Applied to Minimally Invasive Therapy," *IEEE Transactions on Multimedia*, vol. 15, no. 5, pp. 983-1000, 2013. Article (CrossRef Link)

[13] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91-110, 2004. Article (CrossRef Link)

[14] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded Up Robust Features," in *Proc. of European Conference on Computer Vision*, pp. 404-417, 2006. Article (CrossRef Link)

[15] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. of 2011 International Conference on Computer Vision*, pp. 2564-2571, 2011. Article (CrossRef Link)

[16] P. Alcantarilla, J. Nuevo, and A. Bartoli, "Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces," in *Proc. of the British machine Vision Conference*, pp. 13.1-13.11, 2013. Article (CrossRef Link)

[17] T. Tuytelaars and K. Mikolajczyk, "Local Invariant Feature Detectors: A Survey," *Foundations and Trends® in Computer Graphics and Vision*, vol. 3, no. 3, pp. 177-280, 2007. Article (CrossRef Link)

[18] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615-1630, 2005. Article (CrossRef Link)

[19] A. Gruen, "Development and Status of Image Matching in Photogrammetry," in *Proc. of Ian Dowman Retirement Symposium*, vol. 27, no. 137, pp. 36-57, 2012. Article (CrossRef Link)

[20] E. De Castro and C. Morandi, "Registration of Translated and Rotated Images Using Finite Fourier Transforms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, pp. 700-703, 1987. Article (CrossRef Link)

[21] B. S. Reddy and B. N. Chatterji, "An FFT-based technique for translation, rotation, and scale-invariant image registration," *IEEE Transactions on Image Processing*, vol. 5, no. 8, pp. 1266-1271, 1996. Article (CrossRef Link)

[22] X. Tong, K. Luan, U. Stilla, Z. Ye, Y. Xu, S. Gao, H. Xie, Q. Du, S. Liu, X. Xu, and S. Liu, "Image Registration With Fourier-Based Image Correlation: A Comprehensive Review of Developments and Applications," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 10, pp. 4062-4081, 2019. Article (CrossRef Link)

[23] J. P. Lewis, "Fast Template Matching," Vision Interface, Quebec City, QC, Canada, pp. 120-123, 1995.
[24] L. Jing and Y. Tian, "Self-supervised Visual Feature Learning with Deep Neural Networks: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1, 2020. Article (CrossRef Link)
[25] A. Krizhevsky, "Learning multiple layers of features from tiny images," Univ. of Toronto, Toronto, ON, Canada, 2009.
[26] A. Coates, A. Y. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," *Journal of Machine Learning Research*, vol. 15, pp. 215-223, 2011. Article (CrossRef Link)
[27] V. Villena-Martinez, S. Oprea, M. Saval-Calvo, J. Azorin-Lopez, A. Fuster-Guillo, and R. B. Fisher, "When Deep Learning Meets Data Alignment: A Review on Deep Registration Networks (DRNs)," *Applied Sciences*, vol. 10, no. 21, p. 7524, 2020. Article (CrossRef Link)
[28] T. Nguyen, S. W. Chen, S. S. Shivakumar, C. J. Taylor, and V. Kumar, "Unsupervised Deep Homography: A Fast and Robust Homography Estimation Model," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2346-2353, 2018. Article (CrossRef Link)
[29] I. Rocco, R. Arandjelovic, and J. Sivic, "Convolutional Neural Network Architecture for Geometric Matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 11, pp. 2553-2567, 2019. Article (CrossRef Link)
[30] F. L. Bookstein, "Principal warps: thin-plate splines and the decomposition of deformations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 6, pp. 567-585, 1989. Article (CrossRef Link)
[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2015. Article (CrossRef Link)
[32] Z. Chen, Z. Xu, Q. Gui, X. Yang, Q. Cheng, W. Hou, and M. Ding, "Self-learning based medical image representation for rigid real-time and multimodal slice-to-volume registration," *Information Sciences*, vol. 541, pp. 502-515, 2020. Article (CrossRef Link)
[33] S. Miao, Z. J. Wang and R. Liao, "A CNN Regression Approach for Real-Time 2D/3D Registration," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1352-1363, May 2016. Article (CrossRef Link)
[34] J. M. SloanK, A. Goatman, and J. P. Siebert, "Learning Rigid Image Registration - Utilizing Convolutional Neural Networks for Medical Image Registration," in *Proc. of the 11th International Joint Conference on Biomedical Engineering Systems and Technologies*, vol. 2, pp. 89-99, 2018. Article (CrossRef Link)
[35] S. S. M. Salehi, S. Khan, D. Erdogmus, and A. Gholipour, "Real-Time Deep Pose Estimation with Geodesic Loss for Image-to-Template Rigid Registration," *IEEE Transactions on Medical Imaging*, vol. 38, no. 2, pp. 470-481, Feb. 2019. Article (CrossRef Link)
[36] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997. Article (CrossRef Link)
[37] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, 2019. Article (CrossRef Link)
[38] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. of International Conference on Learning Representations*, 2014. Article (CrossRef Link)
[39] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated Residual Transformations for Deep Neural Networks," in *Proc. of 2017 IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, pp. 5987-5995, 2017. Article (CrossRef Link)
[40] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. of 2017 IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, pp. 248-255, 2019. Article (CrossRef Link)

[41] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: analysis and implementation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 881-892, July 2002. Article (CrossRef Link)

[42] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, 1981. Article (CrossRef Link)

Ruoxin Ma received the BS degree in electrical and information engineering from Hunan University in 2019. He is currently pursuing a master degree at the School of Electronics and Information Engineering in Tongji University, Shanghai, China.

Shengjie Zhao (SM'09) received the BS degree in electrical engineering from the University of Science and Technology of China (USTC), Hefei, China, in 1988; the M.S. degree in electrical and computer engineering from China Aerospace Institute, Beijing, China, in 1991; and the PhD degree in electrical and computer engineering from Texas A&M University, College Station, TX, USA, in 2004. He is currently a Dean with the College of Software Engineering, a Professor with the College of Software Engineering and the College of Electronics and Information Engineering, Tongji University, Shanghai, China. In previous postings, he conducted research at Lucent Technologies, Whippany, NJ, USA, and the China Aerospace Science and Industry Corporation, Beijing. His research interests include artificial intelligence, big data, wireless communications, image processing, and signal processing. He is a Fellow of the Thousand Talents Program of China.

Samuel Cheng (S'01–M'04) received the BS degree in electrical and electronic engineering from the University of Hong Kong, the MPhil degree in physics from Hong Kong University of Science and Technology, the MS degree in electrical engineering from the University of Hawaii, Honolulu, HI, and the PhD degree in electrical engineering from Texas A&M University, in 2004. In 2006, he joined the School of Electrical and Computer Engineering, University of Oklahoma and is currently an associate professor. He is a senior member of the IEEE.