Recurrent Affine Transform Encoder for Image Representation

Liu, Letao ; Jiang, Xudong; Saerbeck, Martin; Dauwels, Justin

DOI
10.1109/ACCESS.2022.3150340

Publication date
2022

Document Version
Final published version

Published in
IEEE Access

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 13, 2022, accepted January 30, 2022, date of publication February 9, 2022, date of current version February 22, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3150340

Recurrent Affine Transform Encoder for Image Representation

LETAO LIU[1], XUDONG JIANG[1], (Fellow, IEEE), MARTIN SAERBECK[2], AND JUSTIN DAUWELS[3]

[1]Department of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798
[2]TÜV SÜD Asia Pacific, Singapore 609937
[3]Department of Microelectronics, Delft University of Technology, 2628 Delft, The Netherlands

Corresponding author: Letao Liu (lliu022@e.ntu.edu.sg)

This work was supported in part by the Singapore Economic Development Board Industrial Postgraduate Program under Grant S17-1298-IPP-II.

ABSTRACT This paper proposes a Recurrent Affine Transform Encoder (RATE) that can be used for image representation learning. We propose a learning architecture that enables a CNN encoder to learn the affine transform parameter of images. The proposed learning architecture decomposes an affine transform matrix into two transform matrices and learns them jointly in a self-supervised manner. The proposed RATE is trained by unlabeled image data without any ground truth and infers the affine transform parameter of input images recurrently. The inferred affine transform parameter can be used to represent images in canonical form to greatly reduce the image variations in affine transforms such as rotation, scaling, and translation. Different from the spatial transformer network, the proposed RATE does not need to be embedded into other networks for training with the aid of other learning objectives. We show that the proposed RATE learns the affine transform parameter of images and achieves impressive image representation results in terms of invariance to translation, scaling, and rotation. We also show that the classification performance is enhanced and is more robust against distortion by incorporating the RATE into the existing classification model.

INDEX TERMS Canonical image base, self-supervised learning, representation learning.

I. INTRODUCTION

Achieving invariance between well-posed and misaligned images is a desired property in computer vision and many other imaging domains [1]–[6]. Downstream tasks can benefit from the invariance, such as object recognition [4], [5], simultaneous localization and mapping (SLAM) [7], image registration [8] and defects removal [9]. The study of the image alignment can be traced back to a well-established problem in computer vision [10], [11].

Traditional feature descriptors such as HOG [12] or SIFT [13] measures the correspondence between two images by detecting and matching the local feature, then pruning mismatches using geometric constraints. Those approaches work in many situations but are vulnerable to situations such as intra-class variations and non-rigid deformations. Convolutional Neural Network (CNN) based feature descriptor [14] mitigates this constraint by learning more representative features. However, in many cases, robustly measuring the correspondence between a pair of images is difficult. Such scenarios can have intra-class variability like background, pose, and affine transform. Joint image alignment [15] aims to align a collection of images, which reduces the intra-class variability. [16] first introduced deep learning into unsupervised joint image alignment, where they use network depth-based features to adjust statistics of the specific data being aligned. Recently, Gradient-Aligned Convolution [17] achieves rotation invariance by implementing a prior pixel-level gradient alignment operation before regular convolution. The attention method in [18] combines spatial attention mechanism and channel attention mechanism to reduce the spatial variance of the object. Alternatively, the eigenvector approach of [19] applies a scale and orientation correction for images based on eigenvectors and eigenvalues of the image covariance matrix. In adaptive Gabor convolutional networks [20], the convolutional kernels are adaptively

The associate editor coordinating the review of this manuscript and approving it for publication was Wenbing Zhao.

multiplied by Gabor filters to achieve invariant information extracted from images. The spatial transformer network (STN) [4] is used in [2] to tackle the joint image alignment problem on larger datasets with higher variability. STN [4] can spatially transform the input images by embedding the spatial transformer block into a target network or algorithm. Inspired by STN, Inverse Compositional Spatial Transformer Networks (IC-STN) [21] further enhance the alignment ability of the STN by adopting a recurrent transform strategy. The intuition of the aforementioned approaches is to fulfill the target network's learning objectives, such as classification or object recognition. However, those methods alone do not have a training objective and hence cannot learn the image affine transform parameter independently to perform image alignment.

One of the advantages of STN and IC-STN is the compact network structure, where they only contain a CNN encoder. Unlike STN and IC-STN, other alignment algorithms usually contain many parameters and use complex structures. For example, [22]–[28] use an autoencoder architecture.

In summary, previous approaches for achieving invariance to affine transforms in images have three limitations: i) Some of these algorithms only contain a spatial invariance module [4], [17]–[21] embedded in a neural network designed for classification, object recognition or other tasks. As this module has to be trained via the learning objectives associated with those learning tasks, it is unable to learn the image transform parameters independently. ii) Since this module is trained with the learning objectives other than affine transform, the existing methods only learn the affine transform indirectly or implicitly and hence may not learn the exact affine transform but rather a mixed spatial transform matrix, which may lead to a less optimal affine image alignment. iii) Some of these algorithms require a complicated network structure [22]–[28] instead of a standard CNN.

To address the aforementioned issues, we propose a Recurrent Affine Transform Encoder (RATE) that can be used for unsupervised joint image alignment with compact network architecture. The proposed RATE provides a learning architecture that enables a CNN encoder to learn the affine transform parameter of images. We express an image $\mathbf{X}_o$ as the multiplication of an affine matrix $\mathbf{M}_o$ that describes its pose and a canonical image base $\mathbf{X}_b$. If we purposely transform the image $\mathbf{X}_o$ with a predefined affine matrix $\mathbf{M}$, we obtain another transformed image $\mathbf{X}_t$. Both $\mathbf{X}_o$ and $\mathbf{X}_t$ can be expressed as different transformed versions of the same image $\mathbf{X}_b$. The proposed learning architecture decomposes the predefined affine transform matrix $\mathbf{M}$ into two separate transform matrices $\mathbf{M}_o$ and $\mathbf{M}_t$ to circumvent the problems of unknown ground truth $\mathbf{M}_o$ and $\mathbf{M}_t$ and hence enables unsupervised learning by self-supervision. The proposed RATE is trained by unlabeled image data without any ground truth. The trained RATE infers the affine transform parameter of input images, which can be used to represent images in the canonical form to greatly reduce the image variations in affine transform such as rotation, scaling, and translation, as shown in Figure 1. We refer to Appendix A for a list of abbreviations.

FIGURE 1. Image alignment result on traffic sign dataset. The first row shows variations of a traffic sign image followed by their aligned versions of STN, IC-STN and the proposed RATE shown in the second, third and fourth rows respectively.

FIGURE 2. Block diagram of STN for classification.

In the remainder of the paper, we first review the spatial transformer network (STN) in Section II. We introduce the RATE in Section III, while in Section IV, we show classification performance and image alignment result. We further conduct an in-depth ablation study in Section V. We offer concluding remarks in Section VI.

Contributions: In summary, the main contributions of this paper are: (i) We propose a compact network that can learn the affine transform parameters and infer the canonical image base of the dataset in a self-supervised manner. (ii) We demonstrate through extensive experiments that downstream tasks such as image alignment or classification can benefit from the learned canonical image base.

II. BACKGROUND: SPATIAL TRANSFORMER NETWORK

The intuition of the Spatial Transformer Network (STN) [4] is to fulfil the learning objectives such as image classification or object recognition by spatially transforming the input images. Figure 2 illustrates the STN for classification task, where $\mathbf{X}_o$ is the original input image, E_1 is the spatial transformer block, $\hat{\mathbf{M}}$ is the estimated alignment matrix, $\hat{\mathbf{X}}_b$ is the aligned image, and E_2 is the classification network. The loss function is $\mathcal{L} = \min ||\mathbf{y} - \hat{\mathbf{y}}||_2^2$, where $\mathbf{y}$ and $\hat{\mathbf{y}}$ are the ground truth and predicted image class label.

III. THE PROPOSED RATE

A. TRANSFORM MATRIX FROM AFFINE PARAMETERS

We use variables (x, y) to represent the spatial coordinates of an image $\mathbf{X}$ and define a column vector $\mathbf{x} = (x, y, 1)^T$. Then, an affine transform of an image $\mathbf{X}$ by a transform matrix $\mathbf{M}$ can be expressed by the matrix multiplication $\mathbf{Mx}$. The transform matrix $\mathbf{M}$ mixes different affine parameters together, making the physical meaning of the transform unclear. To represent images with clear affine parameter,

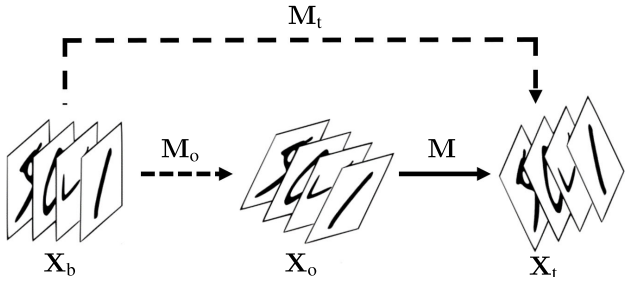

FIGURE 3. Illustration of affine transform decomposition. Solid line refers to the transform from an original input image $\mathbf{X_o}$ to a transformed image $\mathbf{X_t}$. Dashed lines refer to the transform from the canonical image base $\mathbf{X_b}$ to the original image $\mathbf{X_o}$ and to the transformed image $\mathbf{X_t}$ respectively.

we propose to construct the transform matrix $\mathbf{M}$ from affine parameters with a fixed sequence: rotation θ, horizontal and vertical zooms (p, q) and translations (x, y). We denote a set of affine parameters as $\mathbf{c} = \{\theta, p, q, x, y\}$. During training, we can control the affine transform range as rotation $\theta \in [-\varepsilon_\theta, \varepsilon_\theta]$, horizontal and vertical zooms $p, q \in [1 - \varepsilon_{pq}, 1 + \varepsilon_{pq}]$, horizontal and vertical translations $x, y \in [-\varepsilon_{xy}, \varepsilon_{xy}]$.

Considering all possible combinations of affine transform, there are many ways to construct the transform matrix from a set of affine parameters. For illustration purposes, here we construct the transform matrix $\mathbf{M}$ as follows (we mention more ways to construct the transforms in the ablation study in Section V. We also show how to construct the transform matrix $\mathbf{M}$ that includes skew transform in Appendix B):

$$\mathbf{M} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p & 0 & 0 \\ 0 & q & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix}. \quad (1)$$

B. DECOMPOSITION OF AFFINE TRANSFORM

An affine transform links two images before and after the transform but an encoder infers the affine parameter from a single input image. To let the encoder network learn the affine transform, we propose to decompose an affine transform into two parts. We express an image $\mathbf{X}_o$ as the combination of an affine matrix $\mathbf{M}_o$ that describes its pose and a canonical image base $\mathbf{X}_b$, $\mathbf{x}_o = \mathbf{M}_o\mathbf{x}_b$. If we purposely transform the image $\mathbf{X}_o$ with a predefined affine matrix $\mathbf{M}$, we obtain another transformed image $\mathbf{X}_t$ from $\mathbf{x}_t = \mathbf{M}\mathbf{x}_o$. Both $\mathbf{X}_o$ and $\mathbf{X}_t$ can be expressed as different transformed versions of the same image $\mathbf{X}_b$, where $\mathbf{x}_o = \mathbf{M}_o\mathbf{x}_b$ and $\mathbf{x}_t = \mathbf{M}_t\mathbf{x}_b = \mathbf{M}\mathbf{M}_o\mathbf{x}_b$ (see Figure 3).

Thus, from one image, we generate a pair of images $\mathbf{X}_\mathrm{o}$ and $\mathbf{X}_\mathrm{t}$ for training the transform encoder. To map the transform from image space to affine parameters, we encode both $\mathbf{X}_\mathrm{o}$ and $\mathbf{X}_\mathrm{t}$ to affine parameters $\hat{\mathbf{c}}_\mathrm{o}$ and $\hat{\mathbf{c}}_\mathrm{t}$ using a learned encoder E. The estimated affine matrices $\hat{\mathbf{M}}_o$ and $\hat{\mathbf{M}}_t$ are then

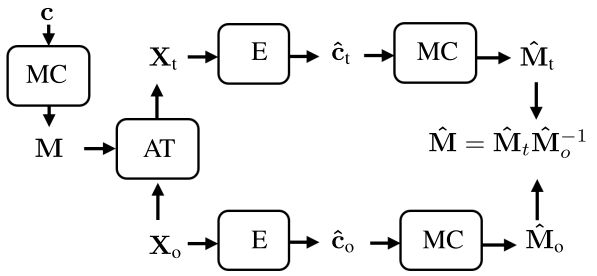

FIGURE 4. Diagram of training the proposed ATE. Inputs: affine parameter $\mathbf{c}$ randomly sampled from the defined transform range and image $\mathbf{X_o}$ sampled from training data. Output: predicted transform matrix $\hat{\mathbf{M}}$. The affine loss is $\mathcal{L}_{\mathbf{affine}} = \min ||\mathbf{M} - \hat{\mathbf{M}}||_2^2$. E stands for Encoder. MC stands for matrix construction. Affine transform (AT) refers to the operation $\mathbf{x_t} = \mathbf{M}\mathbf{x_o}$.

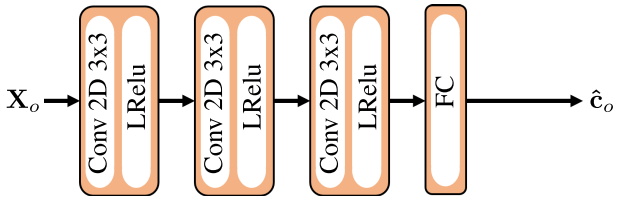

FIGURE 5. Illustration of the encoder network of the proposed RATE. $\mathbf{X_o}$ is the input image, while $\hat{\mathbf{c}}_\mathbf{o}$ is the predicted latent vector.

constructed from $\hat{\mathbf{c}}_\mathrm{o}$ and $\hat{\mathbf{c}}_\mathrm{t}$. The estimated affine matrix $\hat{\mathbf{M}}$ is eventually obtained by $\hat{\mathbf{M}} = \hat{\mathbf{M}}_\mathrm{t}\hat{\mathbf{M}}_\mathrm{o}^{-1}$ (see Figure 4).

The base image $\mathbf{X}_\mathrm{b}$ does not refer to any particular image in the training set, it is rather a learned canonical base of images from the training dataset, where $\mathbf{x}_b = \mathbf{M}_o^{-1}\mathbf{x}_o = \mathbf{M}_t^{-1}\mathbf{x}_t$. It could be the average manifold of all training images within the same category. For instance, the digits "0," "1," ... "9" in MNIST dataset are in different categories. If there are n images of digit "1" with α_i degrees of rotation in the training dataset, $\mathbf{X}_\mathrm{b}$ could be an image of digit "1" with $\sum_{i=1}^{n} \frac{\alpha_i}{n}$ degrees of rotation.

C. LEARNING FRAMEWORK OF THE PROPOSED RATE

The main learning framework of RATE is illustrated in Figure 4. The proposed Recurrent Affine Transform Encoder (RATE) is a small convolutional network with a few convolutional layers and one fully connected layers. The network structure used in this paper is shown in Figure 5 (For different datasets, the exact number of layers and kernel size may vary a bit due to different image sizes). Algorithm 1 describes the procedures to compute the affine regularization loss $\mathcal{L}_\mathrm{affine}$. The loss function of the proposed RATE is $\mathcal{L}_\mathrm{affine} = \min ||\mathbf{M} - \hat{\mathbf{M}}||_2^2$.

D. RECURRENT INFERENCE

Inspired by the recurrent alignment strategy applied in IC-STN [21], we found that the proposed affine transform encoder can also be utilized in a recurrent strategy (RATE) to improve the alignment performance during inference. Figure 6 illustrates the image alignment inference block (AIB) of the proposed RATE. Within the alignment inference

Algorithm 1 Affine Regularizer

Input: training images: $\mathbf{X}_o$, affine parameter: $\mathbf{c}$
Output: $\mathcal{L}_{\text{affine}}$

$\mathbf{M} = \text{Matrix Construction}(\mathbf{c})$
$\mathbf{x}_t = \mathbf{M}\mathbf{x}_o$
$\hat{\mathbf{c}}_t = \text{Encoder}(\mathbf{X}_t)$
$\hat{\mathbf{c}}_o = \text{Encoder}(\mathbf{X}_o)$
$\hat{\mathbf{M}}_t = \text{Matrix Construction}(\hat{\mathbf{c}}_t)$
$\hat{\mathbf{M}}_o = \text{Matrix Construction}(\hat{\mathbf{c}}_o)$
$\hat{\mathbf{M}} = \hat{\mathbf{M}}_t\hat{\mathbf{M}}_o^{-1}$
$\mathcal{L}_{\text{affine}} = \min ||\mathbf{M} - \hat{\mathbf{M}}||_2^2$

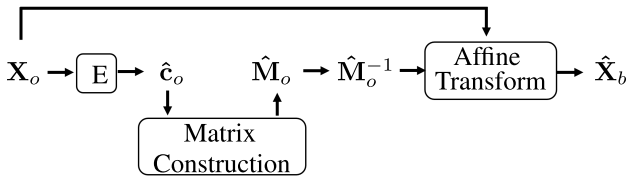

FIGURE 6. Illustration of the image alignment inference block (AIB) of the proposed RATE. $\mathbf{X}_o$ is the input image, E stands for Encoder, $\hat{\mathbf{c}}_o$ is the predicted latent vector, $\hat{\mathbf{M}}_o$ is the predicted transform matrix, $\hat{\mathbf{X}}_b$ is the predicted aligned image.

block (AIB), the estimated latent vector $\hat{\mathbf{c}}_o$ is obtained by feeding the input image $\mathbf{X}_o$ to the encoder E. Upon convergence, $\hat{\mathbf{M}}_o$ obtained from $\hat{\mathbf{c}}_o$ with matrix construction describes the transform from the canonical base $\mathbf{X}_b$ to $\mathbf{X}_o$, and $\hat{\mathbf{M}}_o^{-1}$ describes the transform from $\mathbf{X}_o$ to the canonical base $\mathbf{X}_b$. The estimated canonical base $\hat{\mathbf{X}}_b$ is obtained by applying affine transform on $\mathbf{X}_o$ with matrix $\hat{\mathbf{M}}_o^{-1}$. Figure 7 illustrates the complete image alignment inference process of the proposed RATE. The first estimated canonical image base $\hat{\mathbf{X}}_{b1}$ and estimated transform latent vector $\hat{\mathbf{c}}_o$ are obtained by feeding the input image $\mathbf{X}_o$ to the AIB block. The second estimated canonical image base $\hat{\mathbf{X}}_{b2}$ and estimated transform latent vector $\hat{\mathbf{c}}_{b1}$ are obtained by feeding the first estimated canonical image base $\hat{\mathbf{X}}_{b1}$ to the AIB block again. By repeating this process, we can obtain the estimated canonical image base $\hat{\mathbf{X}}_{bn}$ and estimated transform latent vector $\hat{\mathbf{c}}_{b(n-1)}$. Ideally, $\hat{\mathbf{X}}_{bn}$ is the final result we wish to obtain. However, due to the interpolation effect during affine transform, the resolution of the image becomes lower and lower with more transforms. Hence, we calculate the product of the estimated transform matrix $\hat{\mathbf{M}}_{\text{all}}^{-1} = \hat{\mathbf{M}}_o^{-1}\prod_{i=1}^{n-1}\hat{\mathbf{M}}_{bi}^{-1}$ and then only apply affine transform once on the input image $\mathbf{X}_o$ obtaining aligned image $\hat{\mathbf{X}}_{b(\text{all})}$ to reduce the loss of resolution due to multiple affine transforms.

IV. EXPERIMENTAL RESULTS

In this section, we show the image representation results of the proposed RATE on four datasets: MNIST, SVHN [29], traffic sign [30], scene classification.[1] We also show the image representation result of Tiny-ImageNet [31] with Vision Transformer architecture [32] in Appendix C. We

[1]https://www.kaggle.com/nitishabharathi/scene-classification

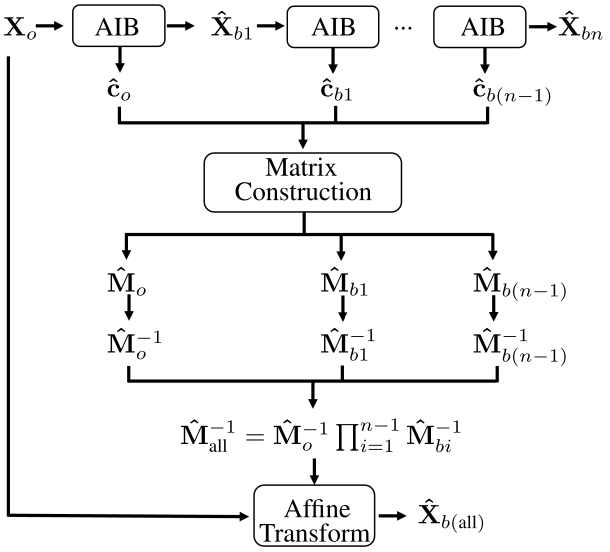

FIGURE 7. Illustration of the complete image alignment inference process of the proposed RATE. AIB refers to the affine inference block in Figure 6.

further show the image representation results of the human face dataset CelebA [33] and CelebA-HQ [34] in Appendix D and E. For quantitative results, we show the classification accuracy from Figure 8 to Figure 11, where we train the classification model (e.g., CNN, STN + CNN, and IC-STN + CNN) with the original dataset without augmentation and test on the dataset with small distortion to examine the robustness of the model. RATE is pretrained with the same range of distortion in a self-supervised manner. We keep the distortion in a small range to simulate the real-world scenarios (e.g., $\varepsilon_\theta = 15°$, $\varepsilon_{pq} = 0.1$ and $\varepsilon_{xy} = 0.1w$, where w is the width of the image). We further divide the distortion range into four quartiles to evaluate the robustness of the model (e.g., "no distortion," "25%," "50%," and "100%"). The comparisons are made between the classification accuracy of the model with and without RATE alignment (e.g., dashed line and solid line). For methods with RATE alignment (e.g., RATE + CNN, RATE + STN + CNN, RATE + IC-STN + CNN), the training and testing images are first aligned by RATE to a canonical image base before being fed to the classifier. For qualitative results, we show the image alignment results with larger distortion (e.g., $\varepsilon_\theta = 45°$) for visualization effect from Figure 12 to Figure 15. We use IC-STN-4 as recommended in the [21] (4 times of recurrent alignment). For RATE, we find it is sufficient to only align the image once for small distortion. We illustrate a detailed recurrent alignment strategy for large distortion (e.g., $\varepsilon_\theta = 180°$) on the human face dataset CelebA in Appendix D. We train all the models using 128 batch size, Adam [35] optimizer with 0.0002 learning rate and 30 epochs. The classification accuracy is the average of 5 runs.

A. MNIST CLASSIFICATION

Due to the simplicity of the MNIST dataset, we use a large range of distortion: scale range $\varepsilon_{pq} = 0.2$ and

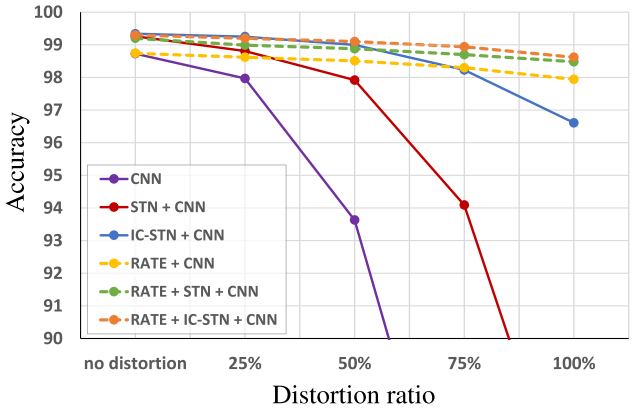

FIGURE 8. Classification accuracy on the MNIST dataset. The notations on horizontal axis refer to the percentage of the maximum distortion (e.g., $\varepsilon_\theta = 15°$, $\varepsilon_{pq} = 0.1$ and $\varepsilon_{xy} = 0.1w$, where w is the width of the image). Dashed and solid lines refer to the methods with and without RATE alignment.

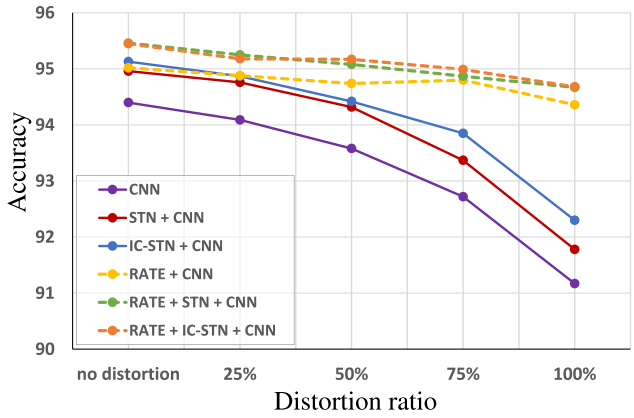

FIGURE 9. Classification accuracy on the SVHN dataset. The notations are the same as in Figure 8.

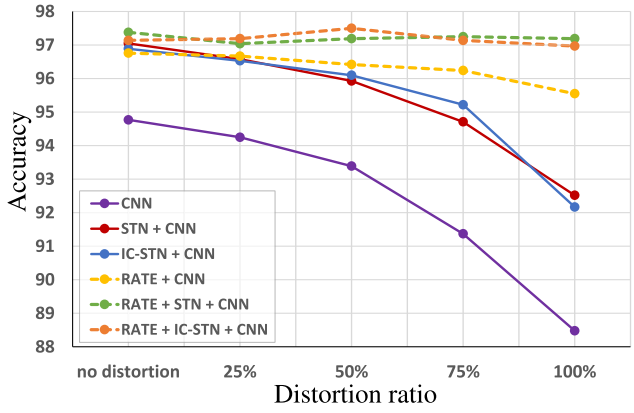

FIGURE 10. Classification accuracy on the GTSRB traffic sign dataset. The notations are the same as in Figure 8.

translation range $\varepsilon_{xy} = 0.2w$. In Figure 8, we observe that the classification accuracy of the models without RATE alignment (solid line) is not robust (curvature of the line) to distortion. However, the performance of IC-STN + CNN is better than STN + CNN and CNN. By contrast, the classification accuracy of the models with RATE alignment (dashed line) is robust (flatness of the line) against distortion. The differences vary less than 1% for all the models with RATE alignment from no distortion to 100% distortion.

B. SVHN CLASSIFICATION

As images in the MNIST are in grayscale with pure background, we further test the proposed RATE on the colored Street View House Numbers (SVHN) dataset [29], which also contains more background variations. As can be seen from Figure 9, the models with RATE alignment are more robust against perturbation. The models with RATE alignment achieve higher accuracy than those without RATE alignment (e.g., RATE + CNN vs CNN) even on test datasets without distortion. RATE + IC-STN + CNN achieves the highest classification accuracy on the test dataset without perturbation and outperforms IC-STN + CNN, which shows that the image alignment strategy of the proposed RATE is also helpful for the classification without distortion.

C. TRAFFIC SIGN CLASSIFICATION

We then evaluate the proposed RATE on the German Traffic Sign Recognition Benchmark (GTSRB) [30], which contains 39,209 training and 12,630 test images from 43 classes taken under various conditions in real-world. The GTSRB dataset is more challenging than the previous ones since some images are taken with motion blur and low resolution (e.g., 15 × 15). In Figure 10, the models with RATE alignment are more robust against distortion. The distortions almost do not affect the classification accuracy of RATE + STN + CNN and RATE + IC-STN + CNN. We also observe that the RATE + CNN outperforms the CNN with a large margin and achieves comparable results with STN + CNN and IC-STN + CNN on the test dataset without distortion.

D. SCENE CLASSIFICATION

Finally, we show how RATE can be applied to higher resolution images in the real world. We evaluate our proposed method on the scene classification dataset, which contains 17,034 training images with 150 × 150 pixels from 6 classes such as buildings, forests, etc. We split the dataset into training and validation datasets with a ratio of 8:2. In Figure 11, the overall classification accuracy is low because we only use a simple CNN as the baseline. The models with RATE alignment achieve higher accuracy than those without RATE alignment (e.g., RATE + CNN vs CNN). RATE + IC-STN + CNN achieves the highest classification accuracy on the test datasets across all distortions and outperforms IC-STN + CNN, which shows that the image alignment strategy of the proposed RATE is helpful for the classification with or without distortion.

E. IMAGE ALIGNMENT RESULTS

In this section, we compare the image alignment results between RATE, STN and IC-STN on images with larger

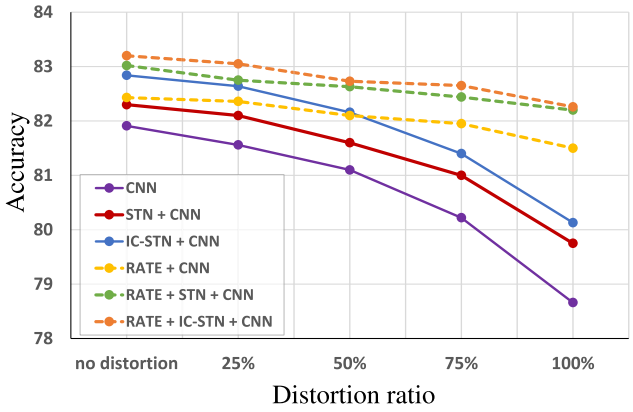

FIGURE 11. Classification accuracy on the Scene dataset. The notations are the same as in Figure 8.

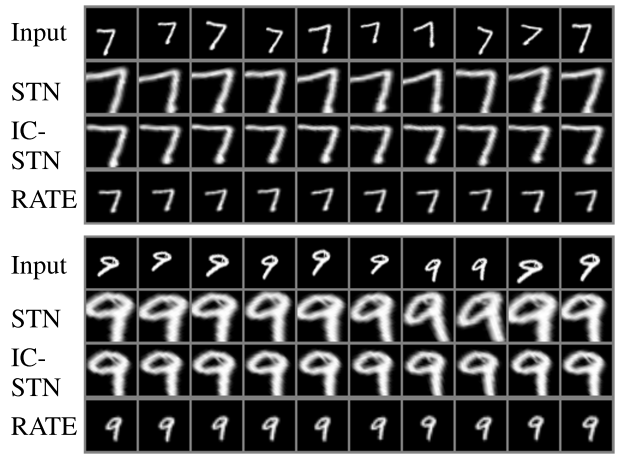

FIGURE 12. Image alignment result on the MNIST dataset. The first row shows variations of a digit image followed by their aligned versions of STN, IC-STN and the proposed RATE shown in the second, third and fourth rows respectively.

FIGURE 13. Image alignment result on the SVHN dataset.

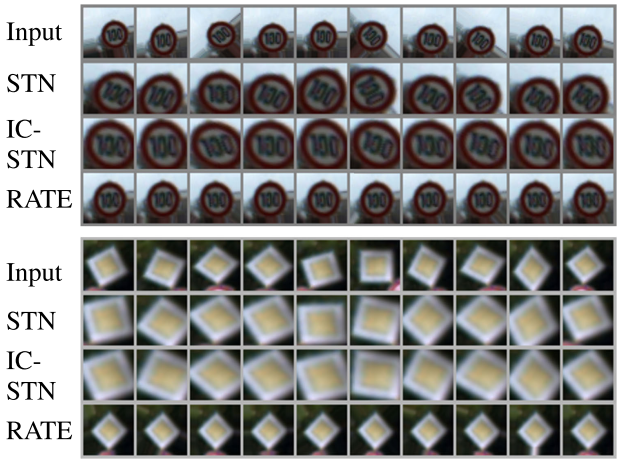

FIGURE 14. Image alignment result on the traffic sign dataset.

distortion for visualization effect, where we increase the rotation to $\varepsilon_\theta = 45°$ (see Figure 12 to 15). All the models are trained using augmented data with the same range as the distortion. The proposed RATE is trained in a self-supervised manner, while STN and IC-STN are trained in a supervised manner. We show more image alignment results on human face dataset CelebA and CelebA-HQ in Appendix D and E and detailed illustrations of the recurrent alignment strategy in Appendix D.

From Figure 12 to 15, the first row shows variations of a sampled image, followed by the aligned versions of STN, IC-STN, and the proposed RATE. Besides the alignment of rotation distortion, we highlight the alignment of the translation in Figure 12 and the scaling in Figure 13 and 14. In Figure 12 and 14, we observe that STN and IC-STN aligned images are larger than the RATE aligned images. This is because the goal of STN and IC-STN is to achieve higher classification accuracy. Hence they tend to make the prominent objects larger. However, the goal of RATE is to align the images to the canonical image base of the dataset. Hence the size of the aligned image obtained by RATE tends to be the average size of the objects in the dataset. For example, the size of the objects aligned by RATE is similar to the average size of the training samples of the same object. Despite the smaller size of the prominent objects aligned by RATE, which may not be in favor of the classification task, models trained with RATE still outperform or achieve comparable results to STN and IC-STN in classification tasks, which shows the effectiveness of image representation of the proposed RATE. We also observe that IC-STN is more robust than STN in terms of rotation variations (see Figure 12 and 13), which may explain why IC-STN tends to outperform STN for classification tasks. Figure 15 shows that the proposed RATE achieves better alignment results compared to STN and IC-STN in more complicated scenarios as well.

F. DEFECTS ALIGNMENT

In this section, we show an application of the proposed RATE: building defects alignment. Buildings may suffer from defects over time, and the inspection photos may be taken

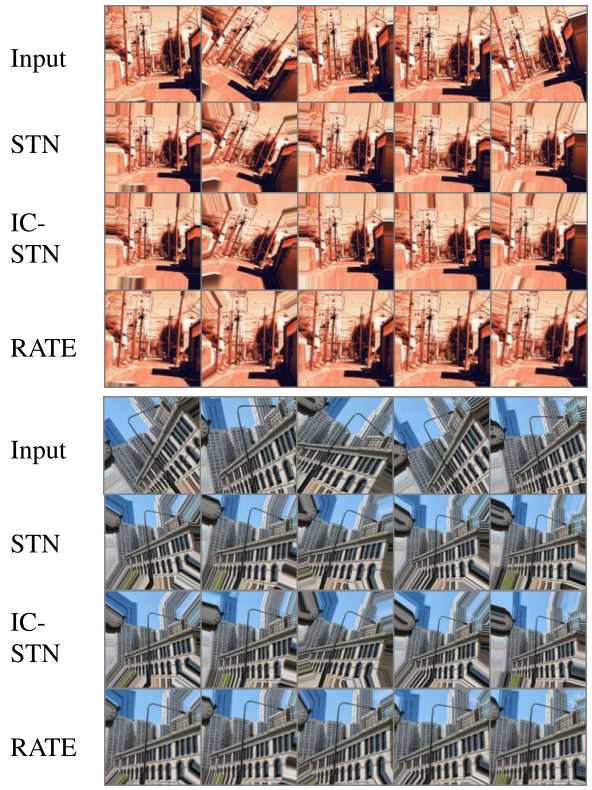

FIGURE 15. Image alignment result on the scene dataset.

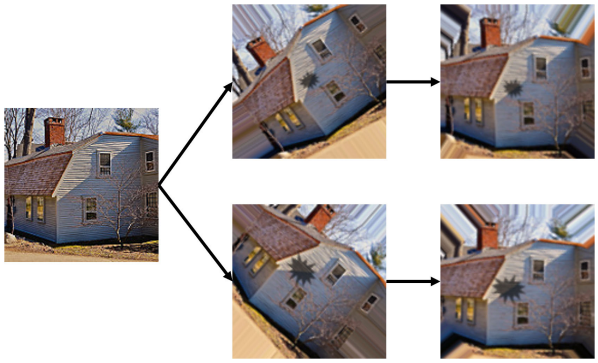

FIGURE 16. Defects alignment result. The image on the left is the building without defects. The first row shows the building with small defects and its aligned pair. The second row shows the building with larger defects and its aligned pair.

from different angles and distances during different trials. Hence, it is difficult to compare whether the defects have become more severe over time based on those photos with different perspectives. Figure 16 illustrates that by using RATE trained on photos with buildings, different photos of the buildings can be aligned to the same canonical pose. Thus, the defects on the building over time can be easily compared.

V. ABLATION STUDY: DIFFERENT AFFINE PARAMETER COMBINATIONS

In this section, we show the effect of learning different affine parameter combinations by the encoder on classification accuracy. We use the RATE + CNN on the traffic sign dataset as the testbed. For Table 1 and 2, the rows show affine parameter combinations learned by RATE, and the columns show the distortion applied on the testing data, the RATE + CNN classifier is trained on data without augmentation. In Table 1, we also include a less commonly used affine transform: skew ("K"). We observe that RATE + CNN performs better than CNN alone. We also observe that the RATE trained with translation learning (RST and RSTK) performs slightly better than that without translation learning (RS).

TABLE 1. Classification error rate of RATE + CNN trained and tested with different affine transforms on the traffic sign dataset. The row CNN is trained without RATE. w/o stands for no distortion on test data.

	w/o	RS	RST	RSTK
CNN	5.2	8.7	11.4	11.9
RS	3.4	3.5	5.9	6.0
RST	3.2	3.6	4.1	4.3
RSTK	3.2	3.6	4.2	4.2

TABLE 2. Classification error rate of RATE + CNN trained and tested with different affine transforms sequence on the traffic sign dataset.

	RST	RTS	SRT	STR	TRS	TSR
RST	4.3	4.7	4.7	4.7	4.4	4.7
RTS	3.9	4.5	4.7	4.1	4.0	4.0
SRT	4.4	4.6	4.6	4.6	4.2	4.5
STR	4.8	4.4	4.5	4.2	4.5	4.6
TRS	4.0	4.2	4.2	4.1	4.2	4.6
TSR	4.7	4.3	4.1	4.2	4.1	4.5

In Table 2, we evaluate the effect of different affine transform sequences on classification accuracy. It shows that the RATE trained with different sequences of affine transforms has similar performance to that trained with the same sequence as in the testing data. This shows that learning different sequences of transforms has little impact on classification.

VI. CONCLUSION

We proposed a Recurrent Affine Transform Encoder (RATE) that can be used for unsupervised image representation learning with compact network architecture. The proposed RATE provides a learning architecture that enables a CNN encoder to learn the affine transform parameter of images and infer the canonical image base of the dataset in a self-supervised manner. The learning architecture decomposes an affine transform matrix into two separate transform matrices to circumvent the problems of unknown ground truth and hence enables unsupervised learning by self-supervision. Downstream tasks such as classification can benefit from the proposed RATE by incorporating it with other models (e.g., CNN, STN + CNN, and IC-STN + CNN). RATE alone can be used for applications such as image alignment and building defect alignment as well. In an in-depth ablation study, we investigated the effect of different combination sequences of affine transforms on classification performance. In future work, we will explore more transforms such as perspective transform and 3D transform.

APPENDIX A
ABBREVIATIONS

All abbreviations used in this paper and their corresponding full name are listed in Table 3.

TABLE 3. Table of abbreviations.

Abbreviation	Full name
RATE	Recurrent Affine Transform Encoder
STN	Spatial Transformer Network
$\mathbf{X}_o$	Original image
$\mathbf{X}_t$	Transformed image
$\mathbf{X}_b$	Canonical image base
$\mathbf{M}_o$	Affine matrix from $\mathbf{X}_b$ to $\mathbf{X}_o$
$\mathbf{M}_t$	Affine matrix from $\mathbf{X}_b$ to $\mathbf{X}_t$
$\mathbf{M}$	Affine matrix from $\mathbf{X}_o$ to $\mathbf{X}_t$
$\mathbf{c}_o$	Affine parameters of $\mathbf{M}_o$
$\mathbf{c}_t$	Affine parameters of $\mathbf{M}_t$
$\mathbf{c}$	Affine parameters of $\mathbf{M}$
MC	Matrix Construction
AT	Affine Transform
AIB	Affine Inference Block

APPENDIX B
AFFINE MATRIX CONSTRUCTION INCLUDING SKEW TRANSFORM

In this section, we show how to construct the affine transform matrix $\mathbf{M}$ that includes the skew transform. The last matrix on the right of Equation 2 is the skew transform matrix.

$$\begin{aligned}\mathbf{M} &= \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ 0 & 0 & 1 \end{bmatrix}, \\ &= \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p & 0 & 0 \\ 0 & q & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix} \\ &\quad \times \begin{bmatrix} 1 & m & 0 \\ n & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \end{aligned} \tag{2}$$

APPENDIX C
RATE WITH VISION TRANSFORMER ON LARGE-SCALE IMAGE DATASET

Apart from conventional CNN architectures, transformer-based architectures [32], [36]–[38] have achieved remarkable performance in visual recognition tasks recently. In this section, we show how to incorporate the proposed RATE with the vision transformer architecture (ViT) [32] and evaluate their performance (ViT vs RATE + ViT) on a large-scale image dataset Tiny-ImageNet [31] both quantitatively (see Figure 18 and qualitatively (see Figure 19).

Tiny-ImageNet [31] is a large-scale image dataset that consists of 100k training images of 200 classes. We evaluate the performance on the validation dataset since the labels for the testing dataset are not available. To obtain a transformer-based classifier on the Tiny-ImageNet, we utilize the ViT [32] (ViT-Base) pretrained on ImageNet-21k [39] as the feature extractor, then remove the pre-trained prediction head and attach a zero-initialized $D \times K_1$ feedforward layer, where $D = 784$ is the input dimension of the MLP head and $K_1 = 200$ is the number of classes in Tiny-ImageNet. The network is fine-tuned based on the training data of Tiny-ImageNet.

To train the RATE for the Tiny-ImageNet, we also utilize the ViT (ViT-Base) pretrained on ImageNet-21k [39] as the feature extractor, then remove the pre-trained prediction head and attach a zero-initialized $D \times K_2$ feedforward layer, where $D = 784$ is the input dimension of the MLP head and $K_2 = 5$ is the number of affine parameters (see Figure 17). The network is then trained on the training data of the Tiny-ImageNet in a self-supervised manner (see Algorithm 1

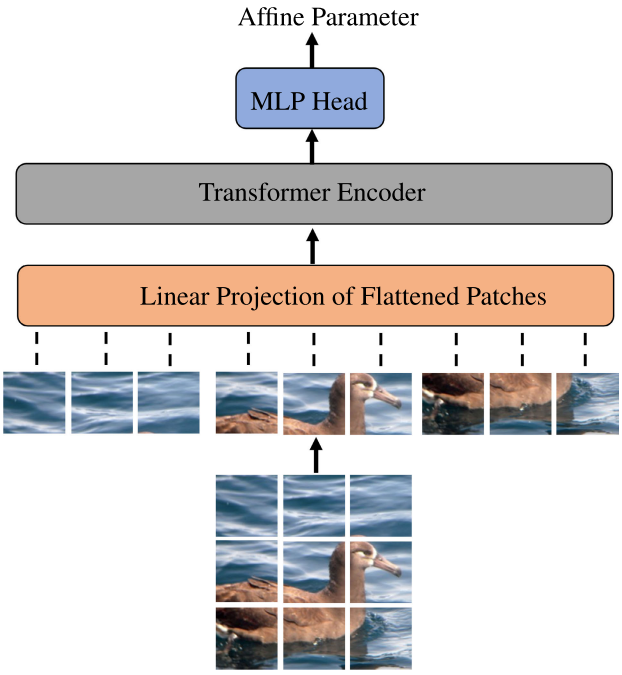

FIGURE 17. RATE incorporated with vision transformer architecture.

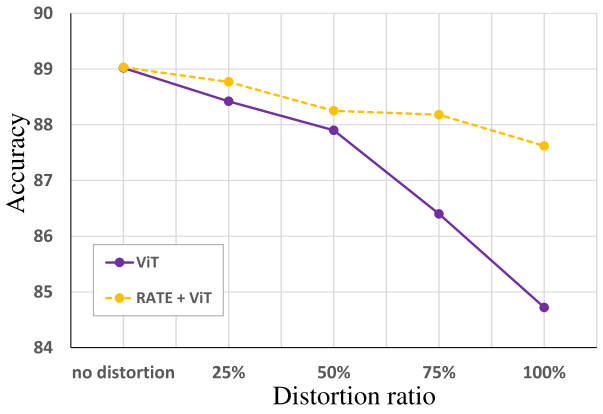

FIGURE 18. Classification accuracy on the Tiny-ImageNet dataset. The notations are the same as in Figure 8.

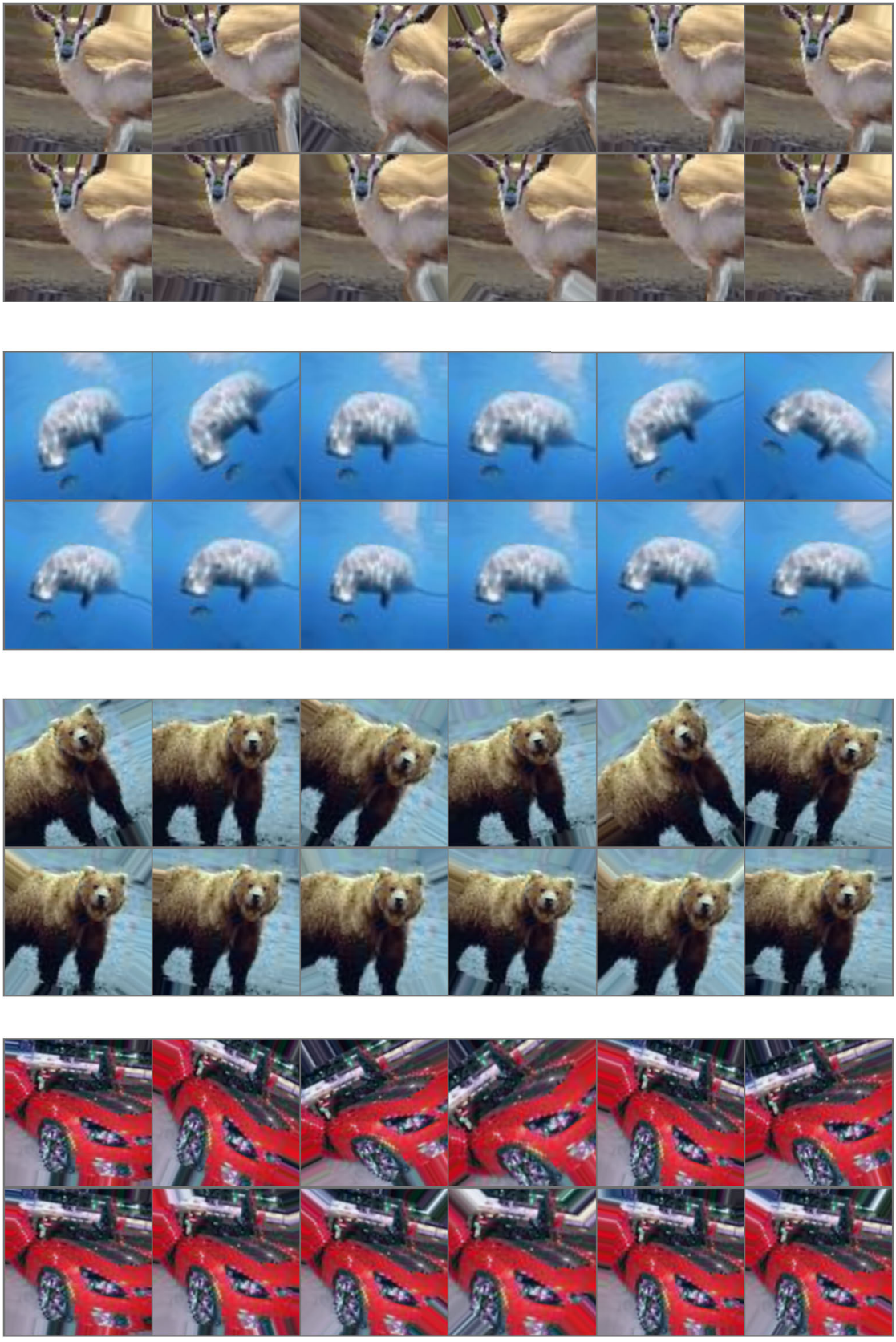

FIGURE 19. Image alignment on the Tiny-ImageNet dataset. The first row shows variations of an image followed by its aligned versions generated by the proposed RATE.

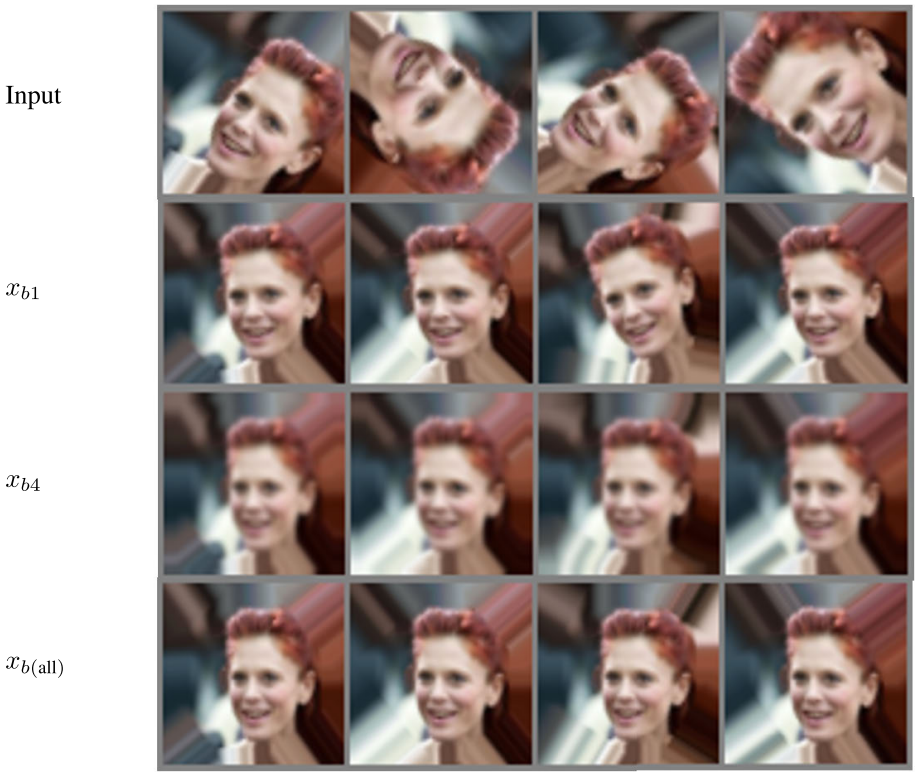

FIGURE 20. Illustration of recurrent image alignment result on CelebA dataset. The first row shows variations of a human face image, followed by the intermediate alignment results x_{b1} and x_{b4}. The last row is the final alignment result $x_{b(\text{all})}$.

FIGURE 21. Illustration of recurrent image alignment result on CelebA dataset. The first row shows variations of a human face image, followed by the intermediate alignment results x_{b1} and x_{b4}. The last row is the final alignment result $x_{b(\text{all})}$.

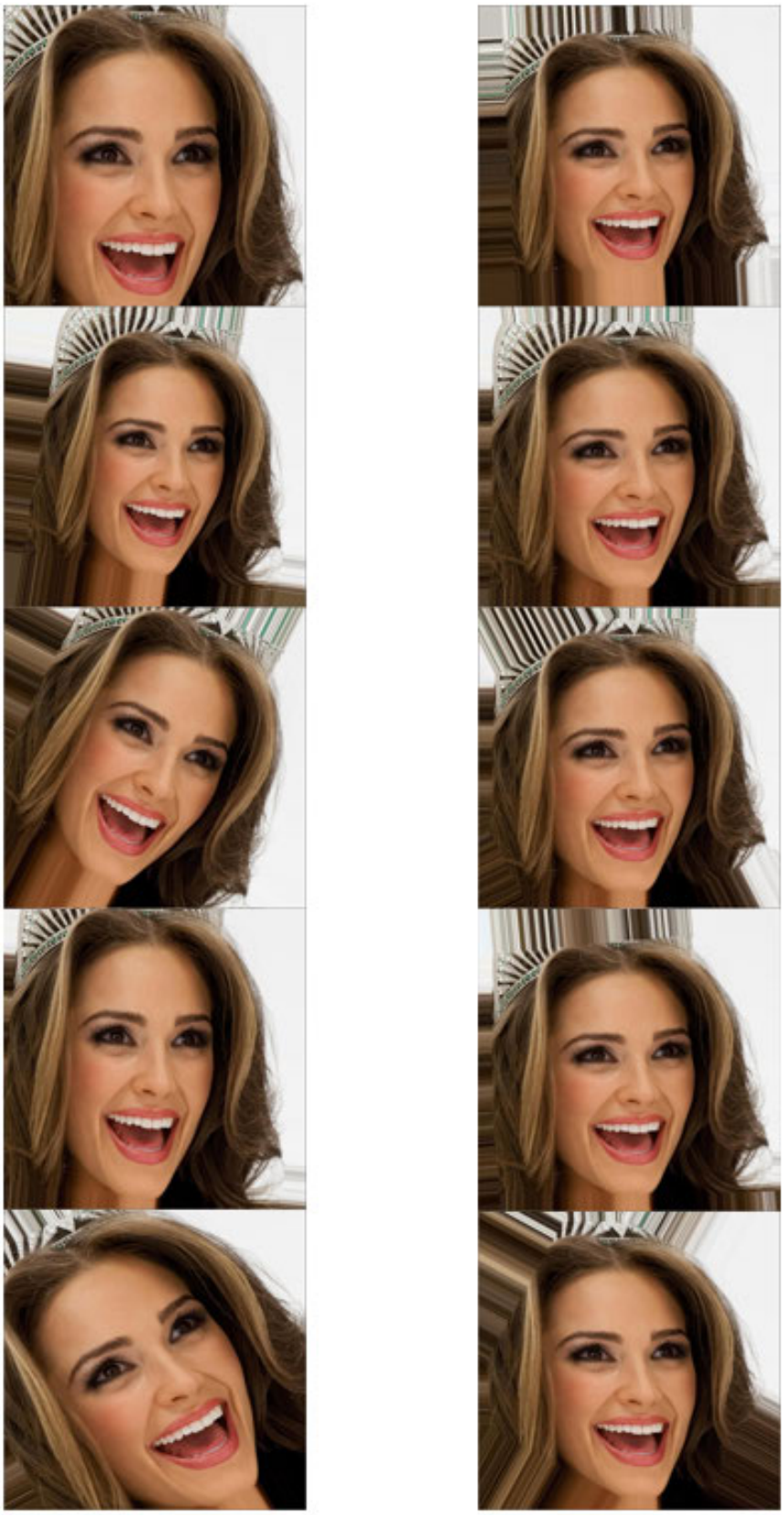

FIGURE 22. CelebA-HQ (1024 × 1024) image alignment results. For each set of images, the left column is the input images, the right column is the aligned images obtained by the proposed RATE.

FIGURE 23. CelebA-HQ (1024 × 1024) image alignment results. For each set of images, the left column is the input images, the right column is the aligned images obtained by the proposed RATE.

and Figure 4). During inference, the affine parameters are estimated from the input image with RATE. The canonical image base $\mathbf{X}_\mathrm{b}$ is then obtained from the input image and the affine parameters. The canonical image base $\mathbf{X}_\mathrm{b}$ is fed to the ViT classifier for classification during training and testing of RATE + ViT.

We utilize the code and the pre-trained model from [40]. We train the models (both RATE and ViT classifier) with 128 batch size, 3000 iterations, and SGD optimizer with a learning rate of 0.003. Due to the limitation of GPU memory, we also set the gradient accumulation step to 3 so that the weights are updated every 3 batches. In Figure 18, we show the classification results of the ViT with and without RATE alignment on testing images of different levels of distortion. To simulate the real-world scenarios, we keep the distortion in a small range from no distortion to the maximum of $\varepsilon_\theta = 15°, \varepsilon_{pq} = 0.1, \varepsilon_{xy} = 0.1w$ (indicated by 100% in Figure 18). The results are the average of 3 trials. As can be seen from Figure 18, the ViT model with RATE alignment is more robust against perturbation compared to ViT model alone. At 100% distortion ratio, the difference between ViT and RATE + ViT is nearly 3 percent, which further verifies the effectiveness of the proposed RATE on image alignment tasks.

In Figure 19, we show the image alignment result of the proposed RATE on the Tiny-ImageNet dataset. For visualization purposes, the rotation range is increased to $\varepsilon_\theta = 45°$. Compared to simple datasets such as MNIST or SVHN, which only contain similar types of objects, Tiny-ImageNet includes a large variety of objects such as animals, vehicles and sophisticated backgrounds. Despite this challenge, Figure 19 suggests that RATE can learn the canonical image base across different types of objects.

APPENDIX D
ILLUSTRATION ON RECURRENT ALIGNMENT STRATEGY

In this section, we show the recurrent image alignment strategy of the proposed RATE on CelebA human face dataset [33]. In Figure 20 and 21, the distortion range is $\varepsilon_\theta = 180°, \varepsilon_{pq} = 0.1, \varepsilon_{xy} = 0.1w$. The first row is the distorted images, followed by the intermediate alignment results x_{b1} and x_{b4}. The last row is the final alignment result $x_{b(\mathrm{all})}$. We observe that x_{b1} can almost align the distorted images to the canonical base. With more iterations of alignment, some tiny distortion is further fine tuned. However, more iterations of affine transform make the images become blurry. Hence, we can combine all the affine transforms and only transform once to obtain $x_{b(\mathrm{all})}$.

APPENDIX E
IMAGE ALIGNMENT RESULTS ON CelebA-HQ

Image samples aligned on the CelebA-HQ dataset (1024 × 1024) [34] are illustrated in Figure 22 and 23. The RATE network for the CelebA-HQ dataset is shown in Table 4. We see that we only need to add a few convolution layers even for high-resolution images like 1024 × 1024.

TABLE 4. The proposed RATE network for CelebA HQ dataset.

Layer	Strides	Act	Output shape
Input Image	-	-	3 × 1024 × 1024
Conv 4 × 4	2	LRelu	16 × 512 × 512
Conv 4 × 4	2	LRelu	32 × 256 × 256
Conv 4 × 4	2	LRelu	64 × 128 × 128
Conv 4 × 4	2	LRelu	128 × 64 × 64
Conv 4 × 4	2	LRelu	256 × 32 × 32
Conv 4 × 4	2	LRelu	512 × 16 × 16
Conv 4 × 4	2	LRelu	512 × 8 × 8
Conv 4 × 4	2	LRelu	512 × 4 × 4
FC	-	Linear	5

REFERENCES

[1] M. Kowalski, J. Naruniec, and T. Trzcinski, "Deep alignment network: A convolutional neural network for robust face alignment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 88–97.

[2] R. Annunziata, C. Sagonas, and J. Cali, "Jointly aligning millions of images with deep penalised reconstruction congealing," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 81–90.

[3] C.-H. Chang, C.-N. Chou, and E. Y. Chang, "CLKN: Cascaded Lucas-Kanade networks for image alignment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2213–2221.

[4] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Advances in Neural Information Processing Systems*, vol. 28, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2015.[Online]. Available: https://proceedings.neurips.cc/paper/2015/file/33ceb07bf4eeb3da587e268d% 663aba1a-Paper.pdf

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, vol. 28, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2015. [Online]. Available: https://proceedings.neurips.cc/paper/2015/file/14bfa6bb14875e45bba028a2% 1ed38046-Paper.pdf

[6] B. Browatzki and C. Wallraven, "3FabRec: Fast few-shot face alignment by reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6110–6120.

[7] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Computer Vision-ECCV* (Lecture Notes in Computer Science), vol. 8690, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham, Switzerland: Springer, 2014, pp. 834–849.

[8] X. Shen, F. Darmon, A. A. Efros, and M. Aubry, "RANSAC-Flow: Generic two-stage image alignment," in *Computer Vision-ECCV*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham, Switzerland: Springer, 2020, pp. 618–637.

[9] C. Lei, X. Huang, M. Zhang, Q. Yan, W. Sun, and Q. Chen, "Polarized reflection removal with perfect alignment in the wild," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1750–1758.

[10] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[11] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. New York, NY, USA: Cambridge Univ. Press, 2003.

[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 886–893.

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2003.

[14] I. Rocco, R. Arandjelovic, and J. Sivic, "Convolutional neural network architecture for geometric matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6148–6157.

[15] E. G. Learned-Miller, "Data driven image models through continuous joint alignment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 2, pp. 236–250, Feb. 2006.

[16] G. Huang, M. Mattar, H. Lee, and E. Learned-Miller, "Learning to align from scratch," in *Advances in Neural Information Processing Systems*, vol. 25, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Red Hook, NY, USA: Curran Associates, 2012. [Online]. Available: https://proceedings.neurips.cc/paper/2012/file/d81f9c1be2e08964bf9f24b1%5f0e4900-Paper.pdf

[17] Y. Hao, P. Hu, S. Li, J. K. Udupa, Y. Tong, and H. Li, "Gradient-aligned convolution neural network," *Pattern Recognit.*, vol. 122, Feb. 2022, Art. no. 108354.
[18] X. Yang, Y. Luo, M. Li, Z. Yang, C. Sun, and W. Li, "Recognizing pests in field-based images by combining spatial and channel attention mechanism," *IEEE Access*, vol. 9, pp. 162448–162458, 2021.
[19] S. V. Chathoth, A. K. Mishra, D. Mishra, and S. G. R. K. Sai, "An eigenvector approach for obtaining scale and orientation invariant classification in convolutional neural networks," *Adv. Comput. Intell.*, vol. 2, no. 1, p. 8, Feb. 2022.
[20] Y. Yuan, L.-N. Wang, G. Zhong, W. Gao, W. Jiao, J. Dong, B. Shen, D. Xia, and W. Xiang, "Adaptive Gabor convolutional networks," *Pattern Recognit.*, vol. 124, Apr. 2022, Art. no. 108495.
[21] C.-H. Lin and S. Lucey, "Inverse compositional spatial transformer networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2568–2576.
[22] G. Wu, M. Kim, Q. Wang, B. C. Munsell, and D. Shen, "Scalable high-performance image registration framework by unsupervised deep feature representations learning," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 7, pp. 1505–1516, Jul. 2016.
[23] X. Guo, E. Zhu, X. Liu, and J. Yin, "Affine equivariant autoencoder," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, vol. 7, Aug. 2019, pp. 2413–2419.
[24] R. Bidart and A. Wong, "Affine variational autoencoders: An efficient approach for improving generalization and robustness to distribution shift," 2019, *arXiv:1905.05300*.
[25] Z. Wan, C. Zhang, Y. Geng, H. Fu, X. Peng, P. Zhu, and Q. Hu, "Cross-view equivariant auto-encoder," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2021, pp. 1–6.
[26] L. Ternes, M. Dane, M. Labrie, G. Mills, J. Gray, L. Heiser, and Y. H. Chang, "ME-VAE: Multi-encoder variational autoencoder for controlling multiple transformational features in single cell image analysis," *bioRxiv*, pp. 1–33, Jan. 2021.
[27] T. Matsuo, H. Fukuhara, and N. Shimada, "Transform invariant auto-encoder," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 2359–2364.
[28] T. Bepler, E. Zhong, K. Kelley, E. Brignole, and B. Berger, "Explicitly disentangling image content from translation and rotation with spatial-VAE," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds. Red Hook, NY, USA: Curran, 2019, pp. 15409–15419.
[29] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop Deep Learn. Unsupervised Feature Learn.*, 2011, pp. 1–9. [Online]. Available: http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf
[30] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The German traffic sign detection benchmark," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Aug. 2013, pp. 1–8.
[31] J. Wu, Q. Zhang, and G. Xu, "Tiny ImageNet challenge," Stanford Univ., Stanford, CA, USA, Tech. Rep. CS 231N, 2017, vol. 7, p. 7.
[32] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent.*, 2021, pp. 1–22. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy
[33] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3730–3738.
[34] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–26. [Online]. Available: https://openreview.net/forum?id=Hk99zCeAb
[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, in (Conference Track Proceedings), San Diego, CA, USA, Y. Bengio and Y. LeCun, Eds., May 2015, pp. 1–15.
[36] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 10012–10022.
[37] A. Srinivas, T.-Y. Lin, N. Parmar, J. Shlens, P. Abbeel, and A. Vaswani, "Bottleneck transformers for visual recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 16519–16529.
[38] Y. Li, T. Yao, Y. Pan, and T. Mei, "Contextual transformer networks for visual recognition," 2021, *arXiv:2107.12292*.
[39] T. Ridnik, E. Ben-Baruch, A. Noy, and L. Zelnik-Manor, "ImageNet-21K pretraining for the masses," in *Proc. 35th Conf. Neural Inf. Process. Syst. Datasets Benchmarks Track*, 2021, pp. 1–20. [Online]. Available: https://openreview.net/forum?id=Zkj_VcZ6ol
[40] *Pytorch Implementation of Vision Transformer*. Accessed: Dec. 30, 2021. [Online]. Available: https://github.com/jeonsworld/ViT-pytorch