Manifold Modeling in Quotient Space:
Learning An Invariant Mapping with Decodability of Image Patches

Tatsuya Yokota* and Hidekata Hontani*

*Nagoya Institute of Technology, Aichi, Japan
*RIKEN Center for Advanced Intelligence Project, Tokyo, Japan

Abstract

This study proposes a framework for manifold learning of image patches using the concept of equivalence classes: manifold modeling in quotient space (MMQS). In MMQS, we do not consider a set of local patches of the image as it is, but rather the set of their canonical patches obtained by introducing the concept of equivalence classes and performing manifold learning on their canonical patches. Canonical patches represent equivalence classes, and their auto-encoder constructs a manifold in the quotient space. Based on this framework, we produce a novel manifold-based image model by introducing rotation-flip-equivalence relations. In addition, we formulate an image reconstruction problem by fitting the proposed image model to a corrupted observed image and derive an algorithm to solve it. Our experiments show that the proposed image model is effective for various self-supervised image reconstruction tasks, such as image inpainting, deblurring, super-resolution, and denoising.

1 Introduction

The non-local similarity (or long-range dependencies) in an image plays a vital role in various vision applications, and the similarity is generally measured by the relations between image patches [4, 47, 42]. Non-local relationships in an image are exactly described by their affinity matrix. The affinity matrix is a matrix whose size is the number of pixels by the number of pixels, and the \((i,j)\)-th element of the matrix describes the strength of the connection between the \(i\)-th and \(j\)-th pixels. A non-local filter (e.g. NLM [4] and BM3D [10, 11]) is a typical framework for image denoising that utilizes the non-local similarity of the images. The (sparse) affinity matrix is usually obtained by the template matching between the image patches. Self-attention structures (e.g. a non-local block [42] and self-attention modules [47]) used in deep learning are closely related to the non-local filtering with an affinity matrix. The self-attention map corresponds to the affinity matrix, and it is usually calculated by the inner product of the channel fibers in feature maps, where each channel fiber of the feature map corresponds to the linear transform of each image patch in the case of convolutional neural networks (CNN).

The image prior hidden in the CNN structure has attracted attention due to the discovery of the deep image prior (DIP) [38, 39] in recent years, and its relation to nonlocal filters has been discussed by Tachella et al.[36]. In [36], Tachella et al. showed that the weight update in the CNN denoiser could be interpreted as twicing [27] (a non-local filter with feedback) in signal processing from the viewpoint of a neural tangent kernel [18].

The existence of similar patches in an image implies the low dimensionality of the manifold of image patches (e.g., manifold model [30, 29], patch-based denoising auto-encoder (AE) [46]). The latent variables encoded from image patches should be useful for the stable evaluation of the non-local similarity in an image. However, local appearances are often degraded by nuisance operations such as noise addition. The latent variables should be invariant against nuisance operations in order for the evaluation to be stable against degradation as the invariance would improve the quality of the image reconstruction.

In the image patches, we can find groups where the included image patches can be obtained by applying a geometric transformation to the template (the same image patch). For example, image...
Figure 1: Image of a quotient space. The equivalence class is a subset of patches that have equivalence
relations. The quotient space is a set of equivalence classes.

Figure 2: Structure of the proposed MMQS. It consists of five parts: patch extraction $\mathcal{H}$, canonical-
ization $\mathcal{P}$, auto-encoder $A_\Theta$, de-canonicalization $\mathcal{P}^{-1}$, and patch aggregation $\mathcal{H}^\dagger$.

patches sampled along the smooth contour of an object in an image are obtained by rotating one
of the sampled image patches. Image patches sampled from a mirror-symmetric object can then be
transformed to each other via flipping. The long-distance similarity evaluated by the latent variables,
which are invariant against geometric transformations, would help represent global image features,
such as rotation- or mirror- symmetric structures.

However, the latent variables that are invariant against geometric transformations have no decod-
ability of images, i.e., the given images cannot be restored from the latent variables. This is because all
image patches in the same group are encoded at the same point in the latent space (degeneracy). To
realize decodability, the degeneracy of the encoder should be avoided. Thus, the image patches in each
equivalence class are represented by a pair of invariant latent variables and their corresponding geo-
metric transformation parameters. The dimensionality of the resultant latent variables is significantly
reduced because the patches in each class are encoded by the same latent variable. This dimensionality
reduction significantly improves the performance of the vision tasks, especially when the number of
image patches is limited or when some portions of image patches are lost or degraded.

In this study, we addressed the incorporation of invariance into patch manifold learning [46] by ap-
plying the concepts of rotation-flip-equivalence classes and quotient spaces. By introducing equivalence
classes as subsets of image patches that have equivalent relations (such as rotation-flip equivalence), a
quotient space is defined as a set of equivalence classes (see Fig. 1). We can see that the latent variables
that represent equivalence classes in a quotient space are invariant with respect to the rotation-flip
manipulation in the original image space. Subsequently, manifold learning is performed in the quotient
space, hence the name: manifold modeling in quotient space (MMQS). Fig. 2 shows the structure of the
proposed MMQS and compares it to manifold modeling in embedded space (MMES) [46]. MMES
is based on three stages: delay embedding (patch extraction), patch AE, and inverse delay-embedding
(patch aggregation). In contrast, canonicalization and de-canonicalization layers are added before and
after the patch AE in the proposed MMQS. In the canonicalization layer, some rotation or flip op-
erations are applied to each input patch to obtain a canonical patch representation. Subsequently,
al all canonical patches are encoded and decoded by the AE, and the non-canonical form is returned
(de-canonicalization).

Our contributions are summarized as follows:

- We extended MMES to MMQS by introducing equivalence classes and quotient spaces to image
patches.

- We established a non-trivial learning algorithm for MMQS and its application to several image reconstruction tasks, such as image inpainting, deblurring, super-resolution, and denoising.
- We employed rotation-flip equivalence relations in MMQS and demonstrated the effectiveness of MMQS for image modeling in computational experiments.

# 2 Proposed method

In this section, we propose a new method for image reconstruction from corrupted images.

## 2.1 Patch extraction from an image

Let us consider \( X \in \mathbb{R}^{I \times J} \) as a matrix that represents a single image of size \( I \times J \). A patch of size \( \sqrt{p} \times \sqrt{p} \) in \( X \) is represented as a vector \( x \in \mathbb{R}^p \). We define a set \( \mathcal{X} \) as follow:

\[
\mathcal{X} := \{x_1, x_2, ..., x_T\},
\]

where each \( x_t \) is a patch extracted from \( X \), and \( T \) is the number of extracted patches.

Practically, the extracted patches can be represented as a matrix as follows:

\[
\mathcal{H}(X) := [x_1, ..., x_T] \in \mathbb{R}^{p \times T},
\]

where the function \( \mathcal{H} : \mathbb{R}^{I \times J} \rightarrow \mathbb{R}^{p \times T} \) represents the patch extraction process. It can also be implemented by a convolutional layer using \( p \) one-hot kernels. The convolutional layer is usually included in typical deep learning platforms, such as TensorFlow and PyTorch. One-hot kernels can be easily generated by reshaping the \((p \times p)\) identity matrix \( I \) into a \((\sqrt{p} \times \sqrt{p} \times p)\) tensor \( I \). Each frontal slice \( I_{...,i} \in \mathbb{R}^{\sqrt{p} \times \sqrt{p}} \) is an independent one-hot kernel. Thus, we have \( \mathcal{H}(X) = \text{reshape}(\text{conv}(X, \mathcal{I})) \), where \( \text{conv}(\cdot, \cdot) \) is a function that represents a convolutional layer, and \( \text{reshape}(\cdot) \) is a function to reshape a tensor into a matrix.

## 2.2 Patch aggregation for image reconstruction

Next, we define the process of patch aggregation for a single image reconstruction. Patch aggregation can be simply thought of as the pseudo inverse of patch extraction. Thus, patch aggregation is a function \( \mathcal{H}^\dagger : \mathbb{R}^{p \times T} \rightarrow \mathbb{R}^{I \times J} \), and it satisfies

\[
\mathcal{H}^\dagger \mathcal{H}(X) = X,
\]

for any \( X \in \mathbb{R}^{I \times J} \). In practice, this operation returns \( T \) patches to their original locations in a single image. The value of the pixel where multiple patches overlap is calculated by its average.

Patch aggregation can be implemented by using a transposed convolutional layer. Let \( \mathbf{1} \) be an all-one matrix of size \( I \times J \), and calculate the following matrix in advance:

\[
D := \text{trconv}(\text{conv}(\mathbf{1}, \mathcal{I}), \mathcal{I}) \in \mathbb{R}^{I \times J},
\]

where \( \text{trconv}(\cdot, \cdot) \) represents a transposed convolutional layer. Each entry of the matrix \( D \) is the number of overlapping patches during image reconstruction. Using the matrix \( D \), the process of patch aggregation can be written as \( \mathcal{H}^\dagger(\cdot) = \text{trconv}(\text{reshape}^{-1}(\cdot), \mathcal{I}) \odot D \), where \( \text{reshape}^{-1}(\cdot) \) is an inverse function of \( \text{reshape}(\cdot) \) and \( \odot \) is an element-wise division operation.

## 2.3 Auto-encoder and manifold learning

The AE plays the most important role in the proposed method. We employ the denoising AE \([40]\). The main function of the denoising AE is manifold learning, and it is expected that the learned manifold will have low dimensionality in order to be robust to additive noise. It is also possible to explicitly enforce the low dimensionality of the manifold by setting the dimensionality of the intermediate layer to be small.
Figure 3: Examples of eight operations for rotation and flip and their corresponding action matrices. A case of $p = 9$. The second, third, and fourth action matrices ($P_2$, $P_3$, and $P_4$) perform clockwise rotation operations of $90^\circ$, $180^\circ$, and $270^\circ$, respectively. The fifth action matrix ($P_5$) performs a flip operation along the vertical direction. We have $P_6 = P_5P_2$, $P_7 = P_5P_3$, and $P_8 = P_5P_4$.

First, the denoising AE can be constructed by the following optimization problem:

$$\min_{\Theta} E_{x, \tilde{x}} \left[ \|x - A_{\Theta}(\tilde{x})\|^2 \right],$$

(5)

where $x \in X \subset \mathbb{R}^p$ is a patch image of size $\sqrt{p} \times \sqrt{p}$ sampled from a set $X$, and $\tilde{x} \sim N(x, I\sigma^2)$ is a noisy patch image corrupted by a zero-mean white Gaussian noise with variance $I\sigma^2$. Solving (5) can be regarded as seeking a compact manifold $M_\Theta$ such that $X \subset M_\Theta \subset \mathbb{R}^p$.

### 2.4 Manifold learning in quotient space

Here, we introduce a quotient space based on equivalence relations. First, we define a set of action matrices as follow:

$$\mathcal{R} := \{ P_1, \ldots, P_K \},$$

(6)

where each $P_k \in \mathbb{R}^{p \times p}$ must be regular (i.e. $P_k^{-1}$ exists). In this study, we employ eight action matrices for rotation and flip (see Fig. 3). The equivalence class of $x \in X$ is given by $[x] := \{ Px | P \in \mathcal{R} \}$. Then, the quotient set $X/\mathcal{R} := \{ [x] | x \in X \}$ can be defined, and we want to seek a compact manifold $Q$ such that $X/\mathcal{R} \subset Q \subset \mathbb{R}^p/\mathcal{R}$. However, $Q$ is a set of equivalence classes, and it is difficult to represent it using a parametric model. Therefore, we consider embedding $X/\mathcal{R}$ into a vector space by canonicalization; a conventional manifold learning is applied to it.

In practice, we consider the following problem:

$$\min_{\Theta} E_{x, \tilde{x}} \left[ \min_{P \in \mathcal{R}} \|Px - A_{\Theta}(P\tilde{x})\|^2 \right],$$

(7)

where $P$ represents a regular matrix that performs the canonicalization of patch $x$. Note that the best $P$ is different for each $x \in X$, but $P^{(i)}x_i = P^{(j)}x_j$ is satisfied when $[x_i] = [x_j]$ because of the min operation. In other words, all patches with equivalence relations are canonicalized as the same patch by the min operation. We call it a canonical auto-encoder (CAE) in this paper. Note that our formulation (7) is a generalization of (5). When we set $\mathcal{R} = \{ I \}$, our model (7) is reduced to the normal denoising AE (5).

Note that the proposed CAE is very different from the normal AE with data augmentation. A normal AE with data augmentation learns common features for all $\{ P_1x, \ldots, P_Kx \}$; however, the proposed CAE adaptively selects only one action matrix from $K$ candidates for each $x$. In other words, data augmentation expands the data distribution; conversely, our model is expected to shrink the data distribution.

### 2.5 Patch manifold learning from a single image

Let us assume that a single image $X$ is given; we consider learning a patch-based AE from $X$ in this section. In other words, the patch extraction from a single image (1) and manifold learning with CAE
(7) are combined as follows:

\[
\min_{\Theta} \mathbb{E}_n \left( \sum_{t=1}^{T} \min_{P(t)} \|P(t)x_t - A_\Theta(P(t)x_t + n)\|_2^2 \right),
\]  

where \( n \sim N(0, I\sigma^2) \) is the white Gaussian noise. The optimization parameters is not only \( \Theta \) of AE, but also \( P(t) \in \mathcal{R} \) for each \( x_t \in \mathcal{X} \). These are summarized as \{\( \Theta, P(1),..., P(T) \)\}.

Especially for the action matrices \{\( P(t) \)\}_{t=1}^{T}, Eq. (8) is a combinatorial optimization problem with \( K^T \) cases. Therefore, it is difficult to find a global solution to this optimization problem. However, if \( \Theta \) is fixed, each action matrix \( P(t) \) can be regarded as an independent optimization parameter, and the optimization problem can be separated into individual discrete optimization problems for \( T \) action matrices. Furthermore, if all the action matrices are fixed, the cost function can be reduced by updating \( \Theta \) using the gradient descent method. Thus, a local solution can be obtained by applying alternating optimizations on \( \Theta \) and action matrices \{\( P(t) \)\}_{t=1}^{T}.

Finally, the proposed alternating optimization algorithm for learning CAE is as follows. For each \( t \), the \( t \)-th action matrix is updated by

\[
P(t) \leftarrow \arg\min_{P \in \mathcal{R}} \|P x_t - A_\Theta(P x_t)\|_2^2.
\]  

For a cost function defined by

\[
\mathcal{L}(\Theta) := \sum_{t=1}^{T} \|P(t)x_t - A_\Theta(P(t)x_t + n_t)\|_2^2,
\]  

where each \( n_t \) is randomly generated by a Gaussian distribution \( N(0, I\sigma^2) \), and the optimization parameter \( \Theta \) is updated using the gradient descent method:

\[
\Theta_{\text{new}} \leftarrow \Theta_{\text{old}} - \eta \frac{\partial \mathcal{L}}{\partial \Theta}(\Theta_{\text{old}}),
\]  

where \( \eta > 0 \) is the learning rate. In practice, various gradient-descent-based algorithms such as Adam [21] are applicable.

### 2.6 Concept of image model

In contrast, Section 2.5 explains a method for learning a patch-based AE from a given single image \( X \). Here, we consider reconstructing an image \( X \) using an image model defined by a patch manifold. Let us assume that a CAE \( A_\Theta \) is given and that it can be considered as an image model:

\[
S_\Theta := \{X \mid \exists x \in \mathcal{X}, \exists k \in \mathcal{K}, \|P_k x - A_\Theta P_k x\|_2^2 \leq \epsilon \},
\]  

where \( \mathcal{K} := \{1, 2, ..., K\} \), and \( \epsilon > 0 \) is a small value. We consider an image model represented by the set \( S_\Theta \) of all possible images \( X \) that satisfy the above condition. This means that all patches of the image \( X \) included in \( S_\Theta \) are around the manifold represented by AE \( A_\Theta \).

An image reconstruction problem using the given model \( S_\Theta \) can be formulated as follows:

\[
\tilde{X} = \arg\min_{X} \|Y - F(X)\|_F^2, \text{ s.t. } X \in S_\Theta,
\]  

where \( Y \) is an observed (usually corrupted) image, and \( F \) is a linear operator that represents the observation model. For example, we set \( F \) as an identity mapping for the denoising task, a masking operator for the inpainting task, a blurring operator for the deblurring task, and a downsampling operator for the super-resolution task.

Combining (12) and (13), the following optimization problem is obtained:

\[
\min_{X} \|Y - F(H^T P^{-1} A_\Theta P H(X))\|_F^2 + \lambda \|P H(X) - A_\Theta P H(X)\|_F^2,
\]  

s.t. \( P(t) = \arg\min_{P \in \mathcal{R}} \|P x_t - A_\Theta P x_t\|_2^2 \),

where \( P([x_1,...,x_T]) := [P(1)x_1,...,P(T)x_T] \), and \( P^{-1}([x_1,...,x_T]) := [P(1)^T x_1,...,P(T)^T x_T] \). This optimization problem can be solved by alternating the optimization with respect to \( X \) and \( P(t) \) for each \( t \).

Finally, the reconstructed image can be obtained by \( \tilde{X} = H^T P_{\text{last}}^{-1} A_\Theta P_{\text{last}} H(X_{\text{last}}) \).
2.7 Self-supervised image reconstruction from a corrupted image

In Section 2.5, we considered learning CAE $\mathcal{A}_\Theta$ from a given clean image $X$. In Section 2.6, we reconstructed an image $X$ from a corrupted image $Y$ by using a given CAE $\mathcal{A}_\Theta$. In this section, we consider a case in which both a clean image $X$ and a CAE $\mathcal{A}_\Theta$ are not given, but only a corrupted image $Y$ is given and its observation model $\mathcal{F}$ is known.

This study aims to simultaneously learn a CAE $\mathcal{A}_\Theta$ and reconstruct an image $X$ from the corrupted image $Y$. We combine the techniques explained in Sections 2.5 and 2.6 and propose a new self-supervised image reconstruction model as follows:

$$
\begin{align*}
\minimize_{X, \Theta} L_X(X, \Theta) := L_{rec}(X, \Theta) + \lambda L_{CAE}(X, \Theta), \\
\text{s.t. } P(t) = \arg\min_{P \in \mathbb{R}} \|P x_t - \mathcal{A}_\Theta P x_t\|_2^2, 
\end{align*}
$$

where

$$
\begin{align*}
L_{rec}(X, \Theta) := \|Y - \mathcal{F}(\mathcal{H}^1 P^{-1} A_\Theta \mathcal{P}_H(X))\|_F^2, \tag{16} \\
L_{CAE}(X, \Theta) := \|\mathcal{P}_H(X) - A_\Theta \mathcal{N}_\sigma \mathcal{P}_H(X)\|_F^2, \tag{17}
\end{align*}
$$

and $\mathcal{N}_\sigma([x_1, ..., x_T]) := [x_1 + n_1, ..., x_T + n_T]$ is an operator for adding Gaussian noise $n_t \sim N(0, \sigma^2 I)$ for all $t \in \{1, ..., T\}$.

Similar to (7) and (14), we solve (15) by alternating the optimization with respect to $(X, \Theta)$ and $\{P(t)\}_{t=1}^T$. For each $t$, the $t$-th action matrix is updated using (9). We update $(X, \Theta)$ by

$$
(X_{\text{new}}, \Theta_{\text{new}}) \leftarrow (X_{\text{old}}, \Theta_{\text{old}}) - \eta \frac{\partial L_X}{\partial (X, \Theta)} (X_{\text{old}}, \Theta_{\text{old}}). \tag{18}
$$

Note that the minimization of the cost function $L_X$ with a balance parameter $\lambda > 0$ is somewhat sensitive, with a value of $\lambda$. When $\lambda$ is too large, it is difficult for the reconstructed image to fit the observed image, and optimization sometimes fails. Following the optimization strategy in [46], we also adjust $\lambda$ based on the balance between $L_{rec}$ and $L_{CAE}$.

3 Related works

Studies on unsupervised (or self-supervised) image modeling using neural network structures have been actively conducted in recent years [38, 39, 17, 32, 34, 23, 22, 3, 44, 7, 1, 46]. Among these, image modeling methods target only the image denoising task. In contrast, in this study, denoising and many other image reconstruction tasks based on arbitrary linear observation models such as inpainting, deblurring, and super-resolution can be solved. The most related studies are DIP [38, 39] and MMES [46]. In particular, MMQS can be characterized as a generalization of MMES.

In addition, the approach using quotient spaces in computer vision has been studied for many years. In [31, 33, 41], the concept of a quotient image that normalizes the lighting conditions to improve the accuracy of face recognition was proposed and studied. In [37], the quotient space was introduced as a generalization of MMES. In [31, 33, 41], a representation of the essential matrices in the study of multiple-view geometry. In [25], a quotient auto-encoder (QAE) was proposed for point cloud processing. The QAE [25] and the proposed CAE share the same concept, but their structures are slightly different. The QAE uses an orbit pooling layer that performs max pooling for all latent representations in the same equivalence class instead of the canonicalization layer in the CAE. Thus, latent (canonical) representation in QAE is a mixture of all latent representations; in contrast, the proposed CAE explicitly selects only one representation from them.

The idea of utilizing rotated and flipped images as an equivalence class can be interpreted as adopting a graph representation of the image if the image is represented by each pixel as a node, the adjacency of the pixels as edges, and all the rotated and flipped images are graph isomorphisms. The measuring similarity between images in a graph representation is generalized as a graph matching problem, such as the graph edit distance [5, 6, 26]. Such an approach naturally introduces a quotient space formulation that essentially treats isomorphic graph data as equivalence classes [9, 14, 15].

Non-local similarity filtering methods that describe the relationships between pixels using the distance in a feature space that does not rely on rotation and flip have also been studied [24, 19, 13].
Note that the proposed approach is different from the minimization of contrastive loss with data augmentation [16, 20, 12, 28, 2, 8]. Since our approach is separation for variant manipulation rather than deletion (as in contrasting learning); the decodability is preserved, even after encoding.

4 Experiments

4.1 Standard behaviors with visualization of patch-manifolds

In this section, we show the standard behavior of the proposed MMQS and compare it to that of MMES. A grayscale image named ‘Barbara’ with a size of 256 × 256 was used for this experiment. 70% or 90% of pixels were randomly removed from ‘Barbara’, thus creating two incomplete images. From the two gray-scaled images with missing pixels, we learned the patch manifolds and reconstructed the images using MMES and MMQS; the results were subsequently compared. Hyperparameters were set to \( \tau = 9 \) and \( \sigma = 0.05 \). The structure of the AE was a multi-layered perceptron (MLP) using the Leaky ReLU as an activation function, and the size of the intermediate layer was \((81, 81, 10, 81, 81)\). The Adam optimizer was employed for weight update while adjusting \( \lambda \) such that \( L_{\text{rec}} \) and \( L_{\text{CAE}} \) were minimized in a well-balanced manner.

Fig. 4 shows the optimization behavior when MMQS reconstructed the image from 70% pixel missing. In the early stages of optimization, the CAE loss decreases rapidly and the reconstruction loss decreases relatively slowly. At this stage, the patch AE may still be trivial and does not reflect the input image. Subsequently, to apply the local patterns of the reconstructed image to the input image, the CAE loss temporarily increases, and manifold learning essentially begins from here. Finally, the overall optimization converges as the learning of the patch manifold converges.

Fig. 5 shows a visualization of what the permutation matrix for the canonicalization of each patch looks like as a result of the optimization. It can be seen that the same permutation matrix is selected for similar patches.
Fig. 6(a) shows the visualization of the patch manifold learned from the image where 70% of the pixels are missing. The dimension of the manifold was 10, and the $9 \times 9$ canonical patches were plotted in 2D space obtained via principal component analysis. In the patch manifold of MMES, the stripe patterns oriented in various directions are mixed, whereas, in the patch manifold of MMQS, the stripe patterns are oriented in the horizontal direction. In addition, comparing the reconstructed images, it can be seen that MMQS outperforms MMES. In particular, MMES fails to restore the horizontal stripes of the scarf located at Barbara’s occipital part, whereas MMQS is able to restore it cleanly.

Fig. 6(b) shows the results for the case where 90% of the pixels are missing. In the patch manifold of the MMES, the vertical stripe patterns are dominant, and the reconstruction result is also affected. In particular, stripe pattern artifacts are present on the face. However, in MMQS, the stripe pattern is not learned, and the reconstruction result is characterized by the mottled pattern of the scarf facing in various directions. Unlike MMES, there are no stripe pattern artifacts on the face. In this case, the recovery accuracy of MMQS for true images is inferior to that of MMES, but this is not a problem for our motivation. MMQS is a softer image model than MMES because there are no restrictions on the orientation of the patches in the reconstructed image. It is expected that image reconstruction with a strongly ill-posed setting, such as 90% of the data missing, will be disadvantageous. Conversely, this result can be considered as evidence that the concept of rotational-flip-equivalence class, which is the key idea in this study, has been realized.
4.2 Analysis of similar patch subsets

Next, we show the results of the \( k \)-means clustering of intermediate representations in an AE trained from an incomplete image where 50% of the pixels are missing by MMES and MMQS. We set the size of the intermediate layers as \((81, 81, 10, 81, 81)\) and extracted the central 10-dimensional representations. \( k \)-means clustering with \( k = 16 \) was applied to both the MMES and MMQS cases.

Fig. 7 shows the results. Clusters of similar patches based on the MMES and MMQS criteria are visualized. MMQS seems to have a clearer structure than MMES. In particular, MMQS contains patch clusters that surround the window frames in the image.

4.3 Color image deblurring

Three color images of sizes \( 256 \times 256 \times 3 \) were prepared and artificially blurred by Gaussian kernel functions with widths of 9, 15, and 31, respectively. We assumed that the Gaussian kernel functions were known in advance, and reconstructed the images using DIP, MMES, and MMQS. The settings of MMQS were \( p = 16 \) and \( \sigma = 0.01 \), and the sizes of the MLP intermediate layers were \((512, 16, 512)\).

Fig. 8 shows the result of color image deblurring. The numerical value written below each image is PSNR / SSIM. It can be observed that MMQS outperforms other methods for both PSNR and SSIM.
Figure 10: Results of image denoising.

Table 1: Quantitative evaluation of the color image reconstruction tasks. The best values are highlighted in bold, and the second-best values are underlined.

| Methods          | Deblurring | Super-resolution | Denoising |
|------------------|------------|------------------|-----------|
| MMQS (prop.)     | 23.28/.7998 | 26.66/.8792      | 27.03/.8526 |
| MMES [46]        | 22.86/.7886 | 26.59/.8806      | 26.93/.8441 |
| DIP [38]         | 22.15/.7680 | 26.18/.8705      | 25.81/.8198 |
| Bicubic          | N.A.       | 23.96/.7952      | N.A.      |
| ZSSR [35]        | N.A.       | 24.84/.8331      | N.A.      |
| real ESRGAN [43] | N.A.       | 23.37/.8236      | N.A.      |
| CBM3D [10]       | N.A.       | N.A.             | N.A.      |
| MCWNNM [45]      | N.A.       | N.A.             | 27.65/.8650 |
|                  |            |                  | 26.36/.8363 |

4.4 Color image super-resolution

Four color images of sizes $256 \times 256 \times 3$ and four color images of sizes $512 \times 512 \times 3$ (8 in total) were prepared and downscaled to $1/4$. We assumed that the linear downscaling operator was known in advance, and upscaled reconstruction was performed using bicubic interpolation, ZSSR [35], real ESRGAN [43], DIP, MMES, and MMQS. Note that real ESRGAN is slightly different from other methods in that it is a method that learns an image prior from a large amount of image data. The settings of MMQS were $p = 36$ and $\sigma = 0.1$, and the sizes of the MLP intermediate layers were $(288, 16, 288)$.

Fig. 9 shows the results of the reconstructed images, and Tab. 1 shows the evaluation results by PSNR and SSIM. In Bicubic interpolation, the image are significantly blurred. Ringing-like artifacts occur in the ZSSR. The image reconstructed by real ESRGAN looks great; however, the contrast of the image is overemphasized. The results evaluated by PSNR and SSIM are considerably inferior to those of the other methods. In DIP, the expression of the diagonal fine edges is not good, especially in the leaves; the tip of the leaf was zigzag. It can be seen that MMES and MMQS are well represented by fine edges, and PSNR and SSIM are also superior to the other methods.

4.5 Color image denoising

Seven images of size $256\times256\times3$ with additive noise ($\sigma = 40$) were prepared for this experiments. We assumed noise variance is known, and reconstruct them by CBM3D [10], MCWNNM [45], DIP, MMES, and MMQS. The settings of MMQS were $p = 36$ and $\sigma = 0.05$, and the sizes of the MLP intermediate layers were $(288, 36, 288)$. In optimization, we adjusted $\lambda$ to keep small CAE loss, and used early stopping like DIP.

Fig. 10 and Tab. 1 show the results. The CBM3D and the proposed MMQS performed the top and second-top reconstruction.
5 Limitations and conclusions

In this study, we introduced equivalence classes based on a set of actions for the image space and proposed a novel approach, MMQS, to learn the manifold of the equivalence classes. As a concrete example of the action set, we adopted eight types of permutation matrices, namely, image rotation and flip, and conducted experiments. The size of the action set could be increased, and various actions other than rotation and flip can also be employed. However, there are generally the following limitations. First, each action must be invertible. Second, the encoding cost increases linearly with respect to the size of the action set. Third, the selection of actions is discrete and not parameterized with continuous variables. Originally, image rotation was an action with a continuous angular parameter, but this time it was discretized with a 90° interval. Future works will include the improvement of scalability of the action set, the establishment of continuous parameterization of actions, improvement of the optimization algorithm, and applications to other tasks.

References

[1] Rushil Anirudh, Suhas Lohit, and Pavan Turaga. Generative patch priors for practical compressive image recovery. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 2535–2545, 2021. 6
[2] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In Proceedings of NeurIPS, pages 15535–15545, 2019. 7
[3] Joshua Batson and Loic Royer. Noise2self: Blind denoising by self-supervision. In Proceedings of ICML, pages 524–533, 2019. 6
[4] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In Proceedings of CVPR, volume 2, pages 60–65, 2005. 1
[5] Horst Bunke. On a relation between graph edit distance and maximum common subgraph. Pattern Recognition Letters, 18(8):689–694, 1997. 6
[6] Horst Bunke and Kim Shearer. A graph distance metric based on the maximal common subgraph. Pattern Recognition Letters, 19(3-4):255–259, 1998. 6
[7] Sungmin Cha, Taeeseon Park, Byeongjoon Kim, Jongduk Baek, and Taesup Moon. Gan2gan: Generative noise learning for blind image denoising with single noisy images. In Proceedings of ICLR, 2021. 6
[8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In Proceedings of ICML, pages 1597–1607. PMLR, 2020. 7
[9] Samir Chowdhury and Tom Needham. Gromov-Wasserstein averaging in a Riemannian framework. In Proceedings of CVPRW, pages 842–843, 2020. 6
[10] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space. In Proceedings of ICIP, volume 1, pages 1–313, 2007. 1, 10
[11] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. IEEE Transactions on Image Processing, 16(8), 2007. 1
[12] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In Proceedings of NeurIPS, pages 766–774, 2014. 7
[13] Sven Grewenig, Sebastian Zimmer, and Joachim Weickert. Rotationally invariant similarity measures for nonlocal image denoising. *Journal of Visual Communication and Image Representation*, 22(2):117–130, 2011. 6

[14] Xiaoyang Guo and Anuj Srivastava. Representations, metrics and statistics for shape analysis of elastic graphs. In *Proceedings of CVPRW*, pages 832–833, 2020. 6

[15] Xiaoyang Guo, Anuj Srivastava, and Sudeep Sarkar. A quotient space formulation for generative statistical analysis of graphical data. *Journal of Mathematical Imaging and Vision*, 63:735–752, 2021. 6

[16] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *Proceedings of CVPR*, volume 2, pages 1735–1742. IEEE, 2006. 7

[17] Reinhard Heckel and Paul Hand. Deep decoder: Concise image representations from untrained non-convolutional networks. *arXiv preprint arXiv:1810.03982*, 2018. 6

[18] Arthur Jacot, Franck Gabriel, and Clement Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Proceedings of NeurIPS*, pages 8571–8580, 2018. 1

[19] Zexuan Ji, Qiang Chen, Quan-Sen Sun, and De-Shen Xia. A moment-based nonlocal-means algorithm for image denoising. *Information Processing Letters*, 109(23-24):1238–1244, 2009. 6

[20] Nikolaos Karianakis, Yizhou Wang, and Stefano Soatto. Learning to discriminate in the wild: Representation learning network for nuisance-invariant image comparison. Technical report, Technical Report, UCLA Computer Science Department, 2013. 7

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[22] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2void-learning denoising from single noisy images. In *Proceedings of CVPR*, pages 2129–2137, 2019. 6

[23] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. In *Proceedings of ICML*, pages 2971–2980, 2018. 6

[24] Yifei Lou, Paolo Favaro, Stefano Soatto, and Andrea Bertozzi. Nonlocal similarity image filtering. In *Proceedings of the International Conference on Image Analysis and Processing*, pages 62–71. Springer, 2009. 6

[25] Eloi Mehr, Andre Lieutier, Fernando Sanchez Bermudez, Vincent Guitteny, Nicolas Thome, and Matthieu Cord. Manifold learning in quotient spaces. In *Proceedings of CVPR*, pages 9165–9174, 2018. 6

[26] Bruno T Messmer and Horst Bunke. A new algorithm for error-tolerant subgraph isomorphism detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(5):493–504, 1998. 6

[27] Peyman Milanfar. A tour of modern image filtering: New insights and methods, both practical and theoretical. *IEEE Signal Processing Magazine*, 30(1):106–128, 2012. 1

[28] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 7

[29] Stanley Osher, Zuoqiang Shi, and Wei Zhu. Low dimensional manifold model for image processing. *SIAM Journal on Imaging Sciences*, 10(4):1669–1690, 2017. 1

[30] Gabriel Peyre. Manifold models for signals and images. *Computer Vision and Image Understanding*, 113(2):249–260, 2009. 1

[31] Tammy Riklin-Raviv and Amnon Shashua. The quotient image: Class based recognition and synthesis under varying illumination conditions. In *Proceedings of CVPR*, volume 2, pages 566–571. IEEE, 1999. 6
[32] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. SinGAN: Learning a generative model from a single natural image. In Proceedings of ICCV, pages 4570–4580, 2019. 6

[33] Amnon Shashua and Tammy Riklin-Raviv. The quotient image: Class-based re-rendering and recognition with varying illuminations. IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(2):129–139, 2001. 6

[34] Assaf Shocher, Shai Bagon, Phillip Isola, and Michal Irani. InGAN: Capturing and retargeting the DNA of a natural image. In Proceedings of ICCV, pages 4492–4501, 2019. 6

[35] Assaf Shocher, Nadav Cohen, and Michal Irani. Zero-shot super-resolution using deep internal learning. In Proceedings of CVPR, pages 3118–3126, 2018. 10

[36] Julian Tachella, Junqi Tang, and Mike Davies. The neural tangent link between CNN denoisers and non-local filters. In Proceedings of CVPR, pages 8618–8627, 2021. 1

[37] Roberto Tron and Kostas Daniilidis. On the quotient representation for the essential manifold. In Proceedings of CVPR, pages 1574–1581, 2014. 6

[38] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In Proceedings of CVPR, pages 9446–9454, 2018. 1, 6, 10

[39] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. International Journal of Computer Vision, 128:1867–1888, 2020. 1, 6

[40] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In Proceedings of ICML, pages 1096–1103, 2008. 3

[41] Haitao Wang, Stan Z Li, and Yangsheng Wang. Generalized quotient image. In Proceedings of CVPR, pages 498–505, 2004. 6

[42] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In Proceedings of CVPR, pages 7794–7803, 2018. 1

[43] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In International Conference on Computer Vision Workshops (ICCVW), 2021. 10

[44] Jun Xu, Yuan Huang, Ming-Ming Cheng, Li Liu, Fan Zhu, Zhou Xu, and Ling Shao. Noisy-as-clean: learning self-supervised denoising from corrupted image. IEEE Transactions on Image Processing, 29:9316–9329, 2020. 6

[45] Jun Xu, Lei Zhang, David Zhang, and Xiangchu Feng. Multi-channel weighted nuclear norm minimization for real color image denoising. In Proceedings of ICCV, pages 1096–1104, 2017. 10

[46] Tatsuya Yokota, Hidekata Hontani, Qibin Zhao, and Andrzej Cichocki. Manifold modeling in embedded space: an interpretable alternative to deep image prior. IEEE Transactions on Neural Networks and Learning Systems, 2021. 1, 2, 6, 10

[47] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In Proceedings of ICML, pages 7354–7363. PMLR, 2019. 1