

© 2023 This manuscript version is made available under the CC-BY-NC-ND 4.0 license <https://creativecommons.org/licenses/by-nc-nd/4.0/>

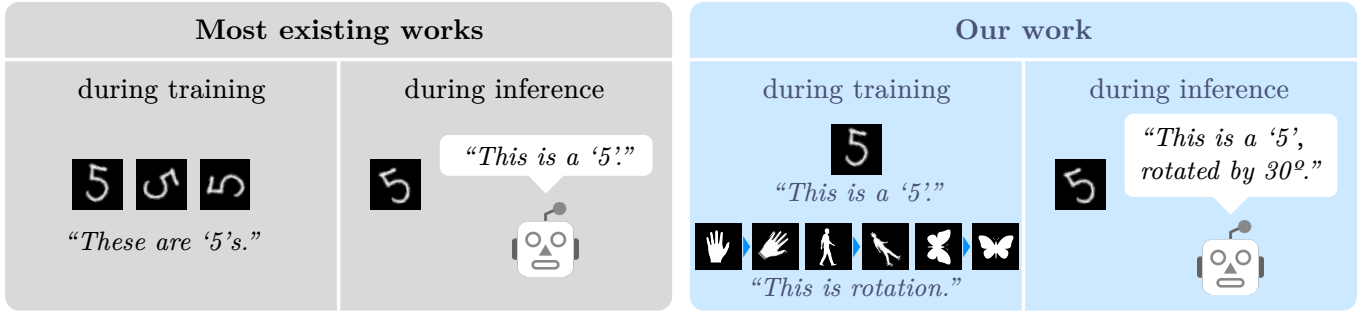
The definitive publisher version is available online at <https://doi.org/10.1016/j.neucom.2023.126882>

# Graphical Abstract

## Toward Extracting and Exploiting Generalizable Knowledge of Deep 2D Transformations in Computer Vision

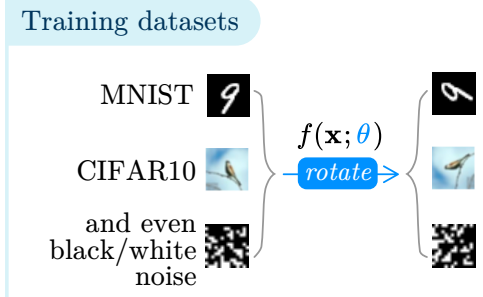
Jiachen Kang, Wenjing Jia, Xiangjian He

### Novelty and Significance

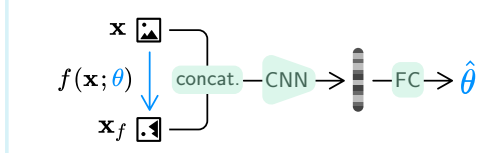


### Methodology and Results

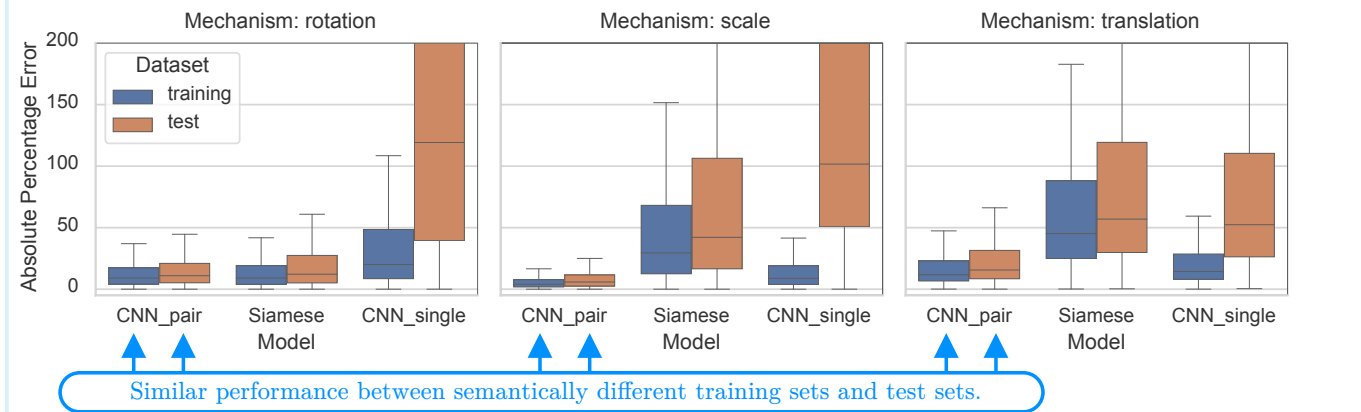
Q1 How to learn generalizable knowledge (e.g. of rotation) ?



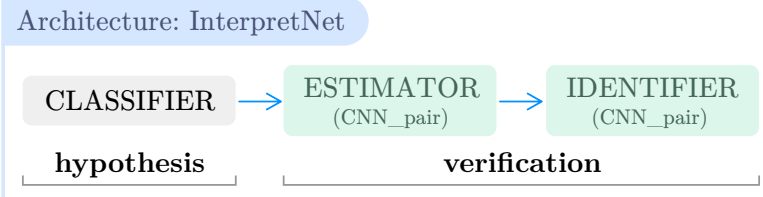
Model: CNN\_pair



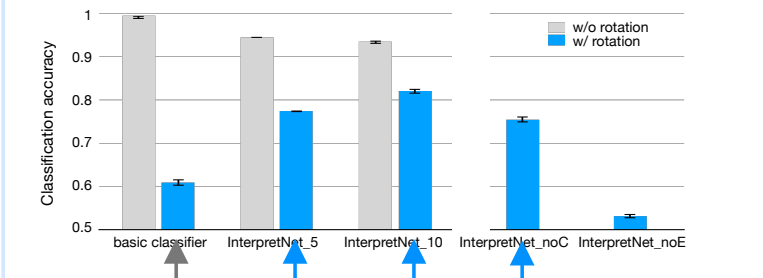
Result: generalizability of learned knowledge



Q2 How to leverage the learned knowledge in classification?



Result: accuracy of classification



Performance improved significantly under covariate shift.

# Highlights

## **Toward Extracting and Exploiting Generalizable Knowledge of Deep 2D Transformations in Computer Vision**

Jiachen Kang, Wenjing Jia, Xiangjian He

- We demonstrate a new learning methodology, with which the Convolutional Neural Networks (CNNs) can learn generalizable knowledge of image transformation mechanisms. Specifically, even if a CNN model is trained on black/white noise, it can still robustly predict the transformation parameters when tested on MNIST, regardless of the domain (semantics) of images.
- We design a novel architecture “InterpretNet” to simulate human visual perception in image classification. With the acquired generalizable knowledge, InterpretNet is able to provide additional explainability in classifying images with covariate shifts. Specifically, in addition to answering questions like “Is there a ‘5’ in the image? ”, the InterpretNet is also able to answer “Why do you think it is a ‘5’ ? ”.

# Toward Extracting and Exploiting Generalizable Knowledge of Deep 2D Transformations in Computer Vision

Jiachen Kang<sup>a</sup>, Wenjing Jia<sup>a,\*</sup>, Xiangjian He<sup>b,\*</sup>

<sup>a</sup>*School of Electrical and Data Engineering, University of Technology Sydney, 15 Broadway, Sydney, 2007, NSW, Australia*

<sup>b</sup>*University of Nottingham Ningbo China, 199 Taikang East Road, Ningbo, 315100, Zhejiang, China*

---

## Abstract

The existing deep learning models suffer from out-of-distribution (*o.o.d.*) performance drop in computer vision tasks. In comparison, humans have a remarkable ability to interpret images, even if the scenes in the images are rare, thanks to the generalizability of acquired knowledge. This work attempts to answer two research questions: 1) the acquisition and 2) the utilization of generalizable knowledge about 2D transformations. To answer the first question, we demonstrate that deep neural networks can learn generalizable knowledge with a new training methodology based on synthetic datasets. The generalizability is reflected in the results that, even when the knowledge is learned from random noise, the networks can still achieve stable performance in parameter estimation tasks. To answer the second question, a novel architecture called “InterpretNet” is devised to utilize the learned knowledge in image classification tasks. The architecture consists of an ESTIMATOR and an IDENTIFIER, in addition to a CLASSIFIER. By emulating the “hypothesis-verification” process in human visual perception, our InterpretNet improves the classification accuracy by 21.1%.

**Keywords:** Deep Learning, Knowledge Acquisition, O.O.D. Generalization, Explainability, Computer Vision

---

## 1. Introduction

Machine learning algorithms based on deep neural networks (DNNs) have made dramatic progress in the field of computer vision in the last decade. Most of these algorithms strongly rely on the assumption of *i.i.d.*, *i.e.*, the training data and test data are independent and identically distributed. In practice, however, the *i.i.d.* assumption can be easily violated due to covariate shift in test datasets [1, 2, 3, 4], which can cause significant performance drops in the models learned from the training set. This is known as the out-of-distribution (*o.o.d.*) generalization problem, which has become one of the main challenges that the deep learning community encounters nowadays. One of the common stopgaps for this problem is to continuously expand the size of training datasets, in order to strengthen the learned invariance of the target objects, by getting rid of other mechanisms (or factors of variation). For example, ImageNet [5], which is a typical dataset for training classification and detection algorithms, contains more than 14 million images. Even so, popular classification models trained with ImageNet have experienced 40 – 45% performance drop when tested on ObjectNet [2], a bias-controlled dataset that produces thousands of images with 600 combinations of parameters, by intervening only on three mechanisms in the photo generation process. This implies that if we try to construct a big enough dataset to approximate the distribution of real-world data, by considering all possible combinations of parameters of mechanisms, the

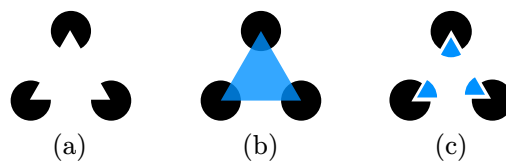


Figure 1: What is in image (a)? There are at least two ways to interpret it, *i.e.*, (b) three black circles partly covered by a white triangle, or (c) three black circles with a notch on each of them. (The former one may have a stronger tendency in perception, according to the Gestalt principles [13].)

number of required data points would be nearly infinite. Similar generalization problems in various sensory domains have been reported in deep learning literature, such as 3D object modeling [6, 7], natural language processing [8, 9], time series signal processing [10, 11, 12], *etc.*

Human beings, in comparison, have powerful *o.o.d.* generalization abilities that enable us to recognize objects based on efficient learning. Extensive studies have shown that learned knowledge can be flexibly reused by infants in novel scenarios [14, 15, 16, 17]. This is analogous to algebraic operations [18], where symbolic variables are manipulated in computational processes. This can be a crucial explanation for the generalization ability. To illustrate this, if we look at Fig. 1(a) [19], based on the same observation, at least two interpretations can be made, as shown in Figs. 1(b) and (c). This simple example illustrates a typical process of human image perception, in which causal inference (in the anti-causal direction) is made by utilizing the mechanisms of either occlusion or notching on variables of circles and/or triangles. Specifically, the process consists of a hypothesis (of the content of three circles and a triangle) and

---

\*Corresponding authors

Email addresses: wenjing.jia@uts.edu.au (Wenjing Jia), sean.he@nottingham.edu.cn (Xiangjian He)

the verification (whether a figure like this can be generated by *covering* the triangle over the circles). If another hypothesis (e.g., of just three circles) and a corresponding verification (by *making a notch* in each of them) can be made, the figure still makes sense to us. This “hypothesis-verification” process in human visual perception has been discussed in detail in [20]. It can be noticed that mechanisms in image generation processes (e.g. occlusion or notching) are crucial in human visual perception. How an image is perceived relies on our knowledge of various mechanisms, rather than knowledge of images that are previously seen (the latter is the way that existing machines operate). It can also be noticed that our knowledge about occlusion or notching is universal and independent of the domain of variables. This generalization is also referred to as systematicity [21]. Based on the above analysis, it can be inferred that it is the generalizability of the knowledge about mechanisms that helps human beings achieve excellent *o.o.d.* generalization performance in visual perception.

While children have plenty of time to gain generalizable knowledge and physical mechanisms through observations and experiments [22, 23, 24], which build foundations for object perception and future knowledge acquisition [25, 26, 16], existing machine learning models rarely have opportunities to do so. One of the main reasons is that current datasets for visual learning inevitably introduce confounding mechanisms, which make it difficult for models to learn unbiased representations and acquire generalizable knowledge [27, 28, 29]. Additionally, most of the studies focus on learning the invariance of objects of interest [30, 31, 32, 33], neglecting the fact that other mechanisms also provide necessary information for perception, as shown in the previous example.

Therefore, in addition to the learning of invariance of target objects, empirical studies are conducted in this paper to learn knowledge about 2D transformation mechanisms (such as rotation, scaling and translation) using DNNs, in order to answer the following questions:

- (1) Whether the learned knowledge of transformation mechanisms can exhibit some degree of generalizability? If so,
- (2) whether the knowledge can be leveraged to facilitate image classification tasks like humans?

In order to answer the first question, it should be made clear what we mean by *the knowledge of a mechanism*. We take the 2D rotation of images as an example of mechanisms. As human beings, if we have learned the knowledge of 2D rotation, it means that for any image, with a proper tool, (a) we can rotate the image at will, and (b) we are able to determine whether (and even how many degrees) the image has been rotated. Obviously, the knowledge that we know about 2D rotation generalizes systematically and is independent of the domain of images. For transformation mechanisms studied in this work, the affine transformation functions are in accord with the description in (a), and are used as a tool to make precise operations<sup>1</sup>. Therefore, our main purpose is the learning of the latter aspect (b),

<sup>1</sup>It does not imply that transformation operations cannot be learned from data. Generative models, which are beyond the scope of this study, have been studied in various tasks [34, 35].

*i.e.*, the estimation of transformation parameters. To achieve this, we devise a new training methodology and use synthetic datasets generated with the target transformation mechanisms for training. It has been found that with this training methodology, the transformation parameters can be estimated accurately and stably, even when networks are trained on random noise and tested on semantically different images.

For the second research question, we propose the architecture of “InterpretNet”, to emulate the *hypothesis-verification* process in human perception in the task of hand-written digit classification, inspired by the theory in [20]. The proposed InterpretNet (Fig. 2) consists of modules of an ESTIMATOR and an IDENTIFIER, which are trained offline separately, and equipped with generalizable knowledge of mechanisms such as 2D transformations. With the acquired knowledge, InterpretNet is able to provide additional explainability in classifying images with covariate shifts. Specifically, in addition to answering questions like “Is there a ‘5’ in the image? ”, the InterpretNet is also able to answer “Why do you think it is a ‘5’? ”. In the case of *o.o.d.* classification task, the test accuracy of InterpretNet is significantly higher than a classic classifier. More impressively, even if it has not seen and thus had no knowledge of hand-written digits during training, InterpretNet can still do the classification, through hypothesis and verification, just like humans.

To the best of our knowledge, InterpretNet is the first work that attempts to learn the generalizable knowledge about mechanisms and use the learned knowledge in image classification. The main contributions are as follows.

- We demonstrate a learning methodology, with which the DNNs can learn generalizable knowledge of image transformation mechanisms robustly using synthetic datasets (and thus answer the first research question).
- We design a novel architecture “InterpretNet” to simulate human visual perception in image classification, with additional explainability, based on the knowledge that has been mastered (and thus answer the second question).

Real-world images can be considered as the result of the interactions between factors of variation, such as foreground and background objects, lighting conditions, camera attributes, *etc.* Additionally, with the rapid development of computer graphics, photo-realistic synthetic datasets with 1) controlled interventions on target factors of variation, and 2) automatic pixel-accurate annotations, can be efficiently created with 3D rendering engines. Therefore, with the proposed methodology and photo-realistic synthetic datasets, target factors of variation can be learned and leveraged in the same manner as in this work.

In the following sections, we first review works related to our study in Section 2. We then propose a novel learning methodology and the architecture “InterpretNet” to learn and leverage generalizable knowledge of 2D Transformations in Section 3. Details of experiments are described, and results are discussed in Section 4. Finally, the paper concludes in Section 5.

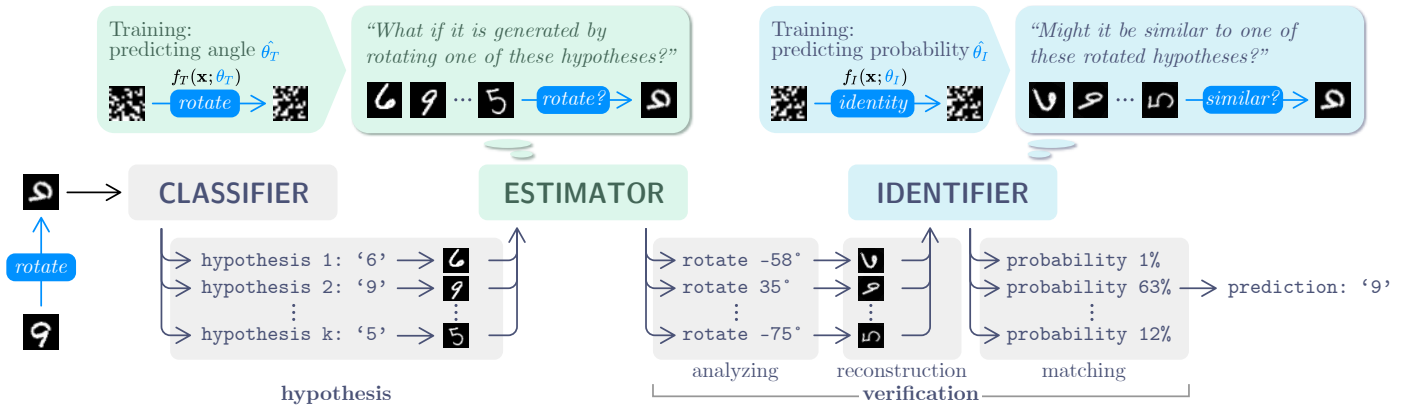


Figure 2: The InterpretNet architecture. Potential classes are hypothesized by the CLASSIFIER  $C$ , and verification on these classes is made by the ESTIMATOR  $E$  and the IDENTIFIER  $I$  through the pipeline of (1) analyzing possible transformations, (2) reconstructing from candidates and (3) matching them with the sample.

## 2. Related Work

In this section, techniques and research works in computer vision related to this work are briefly summarized.

**Data Augmentation and Domain Randomization.** To tackle the potential drop in *o.o.d* performance, effective and commonly used techniques include data augmentation [30, 36, 37, 38, 39] and domain randomization [40, 41].

Data augmentation plays a crucial role in computer vision by expanding the size and diversity of training datasets, reducing overfitting, and enhancing the accuracy of machine learning models. In this section, we briefly review recent works in computer vision to illustrate various data augmentation techniques.

Geometric and color transformations such as rotation, shearing, translation, contrast, brightness, and color jittering are widely used techniques. Researchers often combine these transformations to improve performance. Therefore, Cubuk *et al.* [36] proposed a search space for automated augmentation strategies that control all operations jointly. This technique has led to reduced computational expense and improved performance across various tasks (*e.g.*, 1.0 - 1.3% accuracy improvement on object detection tasks).

Noise injection is another commonly used technique. Kar *et al.* [38] developed an approach that generates noise and corruption by incorporating 3D information consistent with the scene geometry. This approach includes corruptions such as motion blur, fog, *etc.*, which better represents distribution shifts occurring in the real world, leading to a lower error rate across various tasks (*e.g.*, 1.56%  $l_1$  error reduction on the AE benchmark).

Synthetic image generation is gaining attention in computer vision. Hao *et al.* [39] proposed MixGen, a technique that generates new image-text pairs preserving their semantic relationships, thus enhancing data efficiency. This technique achieved significant performance improvements (*e.g.*, a 6.2% accuracy boost on the COCO fine-tuned image-text retrieval task).

Each of these techniques has its own advantages and disadvantages under certain circumstances. For example, in the case of multi-modal pre-training which is growing in influence in computer vision recently, geometric and color transformations

may result in the mismatching of image-text pairs and unnecessary data pollution of multi-modal datasets. Whereas synthetic data generation may be more suitable in such cases, even though they may require additional computational resources.

The technique of domain randomization shares similar principles with data augmentation. While data augmentation is usually referred specifically to as 2D transformations, domain randomization is often adopted when manipulations are made on parameters in 3D environments. From a causal perspective, they both make treatment randomization to get rid of confounders and to improve the learning of invariance. Based on this principle, our work also produces synthetic datasets through treatment randomization, but for a different purpose, that is, instead of randomizing *out* the mechanisms of variation, we aim to take them *into* consideration in classification tasks.

**Parameter Estimation.** As introduced above, the task for learning mechanisms of 2D transformations is to estimate the transformation parameters. This task has been extensively studied in various computer vision topics, such as 2D spatial invariance learning [42], object detection [43, 44], and 3D pose estimations [45, 46], among many others. However, in most of the existing studies, parameter estimation is restricted to object categories that appear in the training sets. An important reason is that single-image parameter estimation is an ill-defined problem, in the sense that parameters of transformations are actually procedural variables, whose values are determined by both of the pre- and post-transformation states. The analysis and results in Sections 3.1 and 4.1.4 show that models trained with methodologies based on single images, can hardly generalize to unseen categories. In this work, the parameter estimation ability that we are interested in should exhibit a certain degree of generalizability similar to humans. Another series of works [47, 48] and the study in [49] conducted representation learning based on pairs of images that are related through mechanisms, by using a single encoder for multiple mechanisms. In this work, to eliminate the potential entanglement from multiple mechanisms, we try to isolate knowledge of single mechanisms and reuse them in downstream tasks.

**Time Series Analysis.** Deep learning studies on time se-

ries cover almost every field of real-world applications, due to its inherent connection with the temporal dimension of the world. These applications include geophysical processes modeling [50], human physical [51, 52] and mental [53, 54, 55] activity analysis, cybersecurity [56], to name a few. This study is related to time series analysis if we consider the transformation of images as a process, and the two most critical time slices are those before and after the transformation, which are of interest to this study. Architectures such as Convolutional Neural Networks (CNNs) [52, 53, 54, 57], Long-Short-Term-Memory (LSTM) [53, 58], Extreme Learning Machine (ELM) [59, 57], *etc.*, are widely used in researches of time series. CNN is adopted in this study to better model 2D image transformations.

**Program Induction.** The knowledge learning in this work is essentially a program induction problem. Active deep learning topics in this area include program synthesis [60, 61], image generation [62, 63], *etc.* Program induction aims for a more effective generation of programs, whereas this work focuses more on the interpretation of images. Therefore, the domain-specific languages in this work are fundamentally different, being more semantically relevant to the downstream tasks.

### 3. Methodology

To answer the two questions raised in the Introduction, we investigate the generalizability of knowledge of image transformation mechanisms and leverage the learned knowledge in image classification tasks. We follow common practices in computer vision community [64, 65], by starting the investigation with the most popular and fundamental MNIST dataset [66]. During image classification, the test set may have a potential covariate shift caused by a target mechanism that cannot be overcome through data augmentation (which is a common situation in real-world applications). We simulate this setting by applying random 2D transformations on the MNIST test set, with no data augmentation operations of any kind performed during training.

Inspired by the human perception process in Fig. 1, we propose that if the machine can learn the knowledge of a target mechanism, it is expected to perform better in the classification tasks under the covariate shift caused by the mechanism. Accordingly, we devise the “InterpretNet” which consists of three DNN modules: a CLASSIFIER  $C$ , an ESTIMATOR  $E$  and an IDENTIFIER  $I$ , to demonstrate this.

We first describe how the datasets are constructed for modules  $E$  and  $I$  to learn generalizable knowledge of mechanisms in Section 3.1. Then we propose the training methodology for modules  $E$  and  $I$  in Section 3.2. InterpretNet makes predictions in classification by raising hypotheses with  $C$  and verifying them with  $E$  and  $I$ . The details about the architecture are described in Section 3.3.

#### 3.1. Synthetic Datasets

To help DNNs learn the generalizable knowledge of a mechanism, the principle, based on which a training set is synthesized, is explained below. Generally, let us denote by  $\mathbf{x}$  and

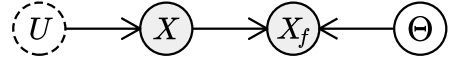


Figure 3: The causal graph of image transformation.  $X$ : Image before the transformation.  $X_f$ : Image after the transformation.  $\Theta$ : Parameter(s) of the transformation in this study, as the variable is randomly sampled, this “treatment randomization” operation removes all arrows pointing to  $\Theta$ .  $U$ : Other unobservable variables that cause the generation of  $X$ .

$\mathbf{x}_f$ , respectively, the images before and after transformation  $f$  (parameterized with  $\theta$ ), then we have

$$\mathbf{x}_f = f(\mathbf{x}; \theta). \quad (1)$$

Note that,  $\theta$  here can be a vector, representing any transformation parameters. In this study,  $\theta$  represents the rotation angle, the scaling factor, the translation offsets, or the combination of the above. As explained in the Introduction, the goal of the knowledge learning is to estimate the value of transformation parameter  $\theta$ . Let  $X$ ,  $X_f$  and  $\Theta$  be the variables from which  $\mathbf{x}$ ,  $\mathbf{x}_f$  and  $\theta$  are instantiated, respectively. According to the causal graph in Fig. 3, if the estimation is made based only on the image *after* transformation, *i.e.*,  $\mathbb{E}(\Theta|X_f)$ , given that  $X_f$  is a collider, conditioning on it will inevitably cause the information flow from  $U$  to  $\Theta$ , which will hinder us from learning stable and thus generalizable knowledge of  $f$  (via  $\Theta$ ). Therefore, in order to remove confounding caused by  $U$ , thus making the prediction of  $\Theta$  generalize better in test domains, we have to condition on both  $X$  and  $X_f$ , *i.e.*, the Markov blanket of  $\Theta$ .<sup>2</sup>

Concretely, in knowledge learning we aim to compute  $\mathbb{E}_{P_{test}}(\Theta|X, X_f)$  given only access to  $P_{train}(\mathbf{x}, \mathbf{x}_f, \theta)$ . The Covariate Shift Assumption and Same Support Assumption, *i.e.*,

$$P_{train}(\theta|\mathbf{x}, \mathbf{x}_f) = P_{test}(\theta|\mathbf{x}, \mathbf{x}_f) \text{ and}, \quad (2)$$

$$supp_{P_{train}}(\mathbf{x}, \mathbf{x}_f) = supp_{P_{test}}(\mathbf{x}, \mathbf{x}_f), \quad (3)$$

are required for the causal model to work, where  $P_{train}$  and  $P_{test}$  are distributions of data in training and test sets, and  $P_{train}(\mathbf{x}, \mathbf{x}_f, \theta) \neq P_{test}(\mathbf{x}, \mathbf{x}_f, \theta)$ .

In this work, synthetic datasets for knowledge learning are constructed according to the above causal framework. Each data point is composed of a pair of images  $\mathbf{x}$  and  $\mathbf{x}_f$  that are before and after the transformation, and the transformation parameter  $\theta$ . Since the labels are automatically generated and no manual annotation is needed, this can be viewed as a self-supervised learning problem.

Learning the knowledge of transformation mechanisms that can be leveraged in classification involves two tasks, *i.e.*, estimating the parameters of the 2D transformation  $f_T(\mathbf{x}; \theta_T)$ , and determining the identity of an image pair defined by the identity function  $f_I(\mathbf{x}; \theta_I)$ . An image  $\mathbf{x}_T$  generated through the 2D transformation function  $f_T$  can be represented as:

$$\mathbf{x}_T = f_T(\mathbf{x}; \theta_T). \quad (4)$$

<sup>2</sup>This is also intuitively true, because it is pointless to ask how a picture has been transformed when no reference is provided.



In this work, we target 2D transformation mechanisms including rotation, scaling and translation, which are implemented using affine transformation functions. For the identity function  $f_I(\mathbf{x}; \theta_I)$ , when  $\theta_I = 1$ , the function returns a same-identity but transformed image  $\hat{\mathbf{x}}_T$ , and any random sample other than  $\hat{\mathbf{x}}_T$  when  $\theta_I = 0$ . Concretely, the identity function is defined by:

$$\mathbf{x}_I = f_I(\mathbf{x}; \theta_I) = \begin{cases} \hat{\mathbf{x}}_T & \text{if } \theta_I = 1; \\ \mathbf{x}'_T & \text{if } \theta_I = 0, \end{cases} \quad (5)$$

where

$$\begin{aligned} \hat{\mathbf{x}}_T &= f_T(\mathbf{x}; \hat{\theta}_T), \\ \mathbf{x}'_T &= f_T(\mathbf{x}'; \hat{\theta}_T). \end{aligned}$$

Here,  $\mathbf{x}'$  is a random sample other than  $\mathbf{x}$ , and  $\hat{\theta}_T$  is the 2D transformation parameter estimated by the ESTIMATOR  $E$  (see Sections 3.2 and 3.3 for the details).

### 3.2. Knowledge Learning

Based on the above synthetic datasets, the ESTIMATOR  $E$  and the IDENTIFIER  $I$  are trained to learn knowledge of 2D transformations  $f_T$  and the identity function  $f_I$ , respectively. Specifically, we employ  $E$  that takes as the input paired images  $\mathbf{x}$  and  $\mathbf{x}_T$  generated from  $f_T$ , to predict the transformation parameters  $\hat{\theta}_T$ . The role of  $I$ , on the other hand, is to learn from  $f_I$  and to predict the probability that a pair of images are of the same identity. In practice, note that the inputs of  $I$  are  $\mathbf{x}_T$  and  $\mathbf{x}_I$  (instead of  $\mathbf{x}$  and  $\mathbf{x}_I$ ).

The mechanism of  $f_T$  is independent of  $f_I$ , and thus  $E$  is optimized first, by minimizing the mean squared error loss  $L_{MSE}$  on  $\theta_T$ .  $I$  is then trained based on datasets generated with  $f_I$  and  $E$ , and optimized by minimizing the binary cross entropy loss  $L_{BCE}$  on  $\theta_I$ . Therefore, the objectives of knowledge learning in this study can be represented as:

$$\arg \min_E L_{MSE}(E(\mathbf{x}, f_T(\mathbf{x}; \theta_T)), \theta_T), \quad (6)$$

$$\arg \min_I L_{BCE}(I(f_T(\mathbf{x}; \theta_T), f_I(\mathbf{x}; \theta_I)), \theta_I). \quad (7)$$

**Knowledge Learning Models** To obtain modules  $E$  and  $I$  that are capable to learn more generalizable knowledge, we investigate a less studied Convolutional Neural Network (CNN) model, which takes *concatenated* image pairs as input (shown in Fig. 4(a)). This model is referred to as ‘‘CNN\_pair’’ in this paper. We also take two commonly studied CNN models that are relevant to this research as baselines, which are the Siamese networks [67] (Fig. 4(b)) and the Vanilla CNN (Fig. 4(c)). The Siamese networks, extensively studied on datasets with intrinsic relations in metric learning and representation learning, also take image pairs as input during training. The vanilla CNN, which is another common method for numerical regression tasks, takes single-images as input and is denoted as ‘‘CNN\_single’’ in this paper.

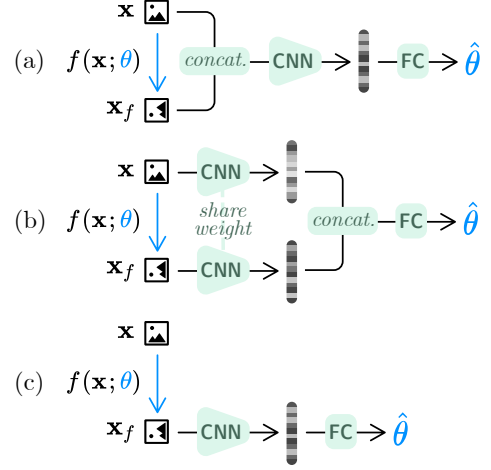


Figure 4: The forward process of three CNN models used for knowledge learning. (a) CNN\_pair: paired images  $\mathbf{x}$  and  $\mathbf{x}_f$  are concatenated in channel dimension before being fed into CNN. The transformation information is encoded as representations in the latent space, which are then sent to the fully connected (FC) layer; (b) Siamese network:  $\mathbf{x}$  and  $\mathbf{x}_f$  are fed into CNN separately. The representations are then concatenated and fed into the FC layer; (c) CNN\_single: Only the transformed images  $\mathbf{x}_f$  are fed into CNN and encoded. The representations are then linearly transformed through the FC layer, and the 2D transformation parameters are predicted as output.

### 3.3. InterpretNet for Classification

With the learned knowledge, we then demonstrate how the learned knowledge can be leveraged for image classification tasks. Towards this end, we design an ‘‘InterpretNet’’ architecture. The InterpretNet consists of three DNN modules: a CLASSIFIER  $C$ , an ESTIMATOR  $E$  and an IDENTIFIER  $I$ . It makes classification predictions by first raising hypotheses with  $C$  and then verifying them with  $E$  and  $I$ . We now describe in detail each module and its roles in simulating the human hypothesis-verification process.

**Classifier  $C$ .** To create an *o.o.d.* task, images in the MNIST test set are transformed before testing, denoted by  $X_T^{test}$ , while those in the training set, denoted by  $X^{train}$ , are original ones without any transformation. Given a test sample  $\mathbf{x}_T^{test} \in X_T^{test}$ , the CLASSIFIER  $C$  produces a probability distribution of the test sample across all classes, which is exploited as confidence scores. If the highest confidence score across all classes is lower than a preset threshold, instead of making a prediction,  $C$  will output a hypothesis  $H(\mathbf{x}_T^{test}) = \{y_i\}_{i=1}^k$ , containing a list of class labels with the top  $k$  confidence scores for further verification.

**Estimator  $E$ .** The ESTIMATOR  $E$  randomly samples  $N(N \geq 1)$  candidates from  $X^{train}$  for each class in  $H(\mathbf{x}_T^{test})$ . Concretely, if the set of all candidates for  $\mathbf{x}_T^{test}$  is denoted by  $X_c \subset X^{train}$ , we have  $X_c = \{X_c^{(y_i)}\}_{i=1}^k$ , and each  $X_c^{(y_i)} = \{\mathbf{x}^{(y_i), j}\}_{j=1}^N$ . With the assumption that  $\mathbf{x}_T^{test}$  may be transformed from what looks similar to some of the candidates in  $X_c$ ,  $E$  then analyzes the relationship between  $\mathbf{x}_T^{test}$  and each candidate *w.r.t.* the 2D transformation using the knowledge learned previously, by computing  $\hat{\theta}_T^{i,j} = E(\mathbf{x}^{(y_i), j}, \mathbf{x}_T^{test})$ .

**Identifier  $I$ .** Since  $E$  is a deterministic function and will produce an output regardless of whether two images are really related, the role of the IDENTIFIER  $I$  is to examine which candidate



is more similar to the  $\mathbf{x}_T^{test}$ . To achieve this, firstly, reconstructions are performed on each candidate by exploiting the parameters  $\widehat{\theta}_T^{i,j}$  predicted by  $E$  and we obtain  $\widehat{\mathbf{x}}_T^{(y_i),j} = f_T(\mathbf{x}^{(y_i),j}; \widehat{\theta}_T^{i,j})$ . Then,  $\widehat{\mathbf{x}}_T^{(y_i),j}$  is tested with  $I$  on how likely it matches to  $\mathbf{x}_T^{test}$ , using  $I(\mathbf{x}_T^{test}, \widehat{\mathbf{x}}_T^{(y_i),j})$ , which is trained with the identity function  $f_I$ . The label of the candidate with the highest likelihood will be output as the final prediction  $\hat{y} = \arg \max_{y_i} I(\mathbf{x}_T^{test}, \widehat{\mathbf{x}}_T^{(y_i),j})$ .

In the above process, the potential classes are hypothesized by  $C$ , and verification on these classes is made by modules  $E$  and  $I$  through the pipeline of (a) analyzing possible transformations, (b) reconstructing from candidates, and (c) matching them with the sample.

It can also be noticed that the pre-trained modules  $E$  and  $I$  do not have access to MNIST during training, and do not have to rely on  $C$  either. Based on the fact that the training and test set of MNIST share the same class label space, we also explore the architecture of InterpretNet without a  $C$  (denoted as ‘‘InterpretNet\_noC’’). The only difference is that InterpretNet\_noC directly takes all classes as a hypothesis ( $k = 10$ ).

The codes of our methodology are publicly available<sup>3</sup>.

## 4. Experiments

In this section, experiments are conducted to answer the two questions raised in the Introduction, *i.e.*,

- (1) whether the learned knowledge of transformation mechanisms can exhibit some degree of generalizability? (Section 4.1)
- (2) whether the knowledge can be leveraged to facilitate image classification tasks like humans? (Section 4.2)

### 4.1. Is the Learned Knowledge Generalizable?

In order to study the robustness of the estimations on  $\theta_T$  of  $f_T$  and  $\theta_I$  of  $f_I$ , synthetic datasets are constructed according to the procedure described in Section 3.1. Three DNN models are trained and tested based on the methodology illustrated in Section 3.2. Next, we first examine the performance of CNN\_pair in Sections 4.1.2 and 4.1.3. The comparisons between the CNN\_pair and the two baseline models, *i.e.*, the Siamese networks and the CNN\_single, are conducted in Section 4.1.4.

#### 4.1.1. Training

**Datasets.** In the experiments, the original images in MNIST, EMNIST [68] and CIFAR-10 [69] are randomly transformed before being used as  $\mathbf{x}$  to alleviate the potential overfitting. We obtain  $\mathbf{x}_T = f_T(\mathbf{x}; \theta_T)$ , where the transformation parameters  $\theta_T$  are randomly sampled in a uniform distribution (see Table 1).

In this work, we conduct learning on four types of  $f_T$ , including individual transformations of rotation, scaling and translation, and the joint transformation of the above three. For learning individual transformations, only one of the three transformations is applied at a time, while all three transformations are applied simultaneously in the joint case.

Table 1: The parameters of 2D transformations investigated in experiments. Each parameter is uniformly sampled within its ranges.

Parameter	Range
Rotation angle	$[-90^\circ, 90^\circ]$
Translation (horizontal)	$[-5, 5]$ pixels
Translation (vertical)	$[-5, 5]$ pixels
Scale factor	$[0.7, 1.3]$

Furthermore, to demonstrate that the generalizable knowledge is independent of the domain of images, a synthetic dataset composed of black/white noises (of a Bernoulli distribution) is randomly generated and used as  $\mathbf{x}$ . To better test generalizability, all test data are sampled from datasets that are semantically different from the training sets. Three groups of experiments are conducted, whose detailed schemes are listed in Table 2.

**Model Settings for Knowledge Learning.** The CNN model in [47] is used as the backbone in CNN\_pair and the two baselines (*i.e.*, the Siamese network and CNN\_single). All input pairs of  $\mathbf{x}$  and  $\mathbf{x}_T$  are concatenated along the channel dimension before being fed into the CNN\_pair. Thus, the input dimension is  $N_{batch} \times 2 \times 28 \times 28$  in Exp\_MNIST and Exp\_NOISE, and  $N_{batch} \times 6 \times 32 \times 32$  in Exp\_CIFAR, where  $N_{batch}$  is the batch size. We keep the default settings for the baselines.

**Training Details.** The CNN models are trained using Adam optimizer with a batch size of 512 and the weight decay set to  $5.0 \times 10^{-4}$ . In Exp\_MNIST and Exp\_CIFAR, the models are trained for 500 epochs in each experiment, with the learning rate initialized to 0.03 and decaying by a factor of 0.6 for every 50 epochs. In Exp\_NOISE, since the noise images are generated on-the-go, the models are trained for  $1.0 \times 10^5$  steps with the same batch size of 512. The initial learning rate is also set to 0.03 with a decaying factor of 0.5, and a decaying cadence of  $1.0 \times 10^4$  steps.

#### 4.1.2. Learning of 2D Transformation Mechanisms

The performance of CNN\_pair on learning the knowledge of the three individual transformations is presented in Figs. 5 and 6. It can be observed in Fig. 5 that the majority proportions of the absolute percentage errors (APE) (*e.g.* the third quartile in the distributions) are below 20% in most experiments for CNN\_pair. Moreover, Due to the domain shift between the training and test sets, varying degrees of distribution shifts of the APE can be observed in Fig. 6. However, the shift is significantly smaller for model CNN\_pair compared to the other two models, namely Siamese network and CNN\_single. Comparing the mean median of the distribution shift across all three mechanisms, the CNN\_pair exhibits a much lower shift of 2.5% APE between training and test sets, while the Siamese network and the CNN\_single exhibit shifts of 9.2% and 76.8%, respectively.

The above prediction performance and the minor distributional difference of APE indicates the robust generalizability of 2D transformation knowledge learned by CNN\_pair. This is a noteworthy finding, considering the fact that the data in the training and test sets differ completely in terms of semantics. More results on the performance of CNN\_pair on 2D transfor-

<sup>3</sup>Codes have been released at <https://github.com/xxx>

mation learning can be found in Appendix A.

#### 4.1.3. Learning of the Identity Function

To evaluate the generalization performance of the  $\theta_I$  estimation, Exp\_NOISE in Table 2 is chosen to train the CNN\_pair because it is the most difficult among the three experiments. Synthetic data is generated for training with the method in Section 3.1, where the parameter  $\theta_I$  randomly takes a value of either 0 or 1 in order to produce a balanced dataset. The resultant F1 scores are 0.9987 and 0.9757 for training and test set, respectively. This minor difference between the two F1 scores again indicates the strong generalizability of knowledge of the identity function learned with CNN\_pair.

#### 4.1.4. Key Elements in Knowledge Learning

In this section, ablation studies are conducted to examine elements crucial for learning generalizable knowledge.

Firstly, as analyzed based on the causal graph in Fig. 3, if there exists a causal relationship from  $U$  to  $X$ , it is necessary to condition on both  $X$  and  $X_f$  in order to predict  $\Theta$  robustly. As shown in Fig. 6, the performance drop of the generalization of CNN\_single is much more severe in all learning cases, compared with CNN\_pair and Siamese networks that both take paired images  $X$  and  $X_f$  as inputs. The translation learning of CNN\_single generalizes relatively better than rotation or scaling learning of CNN\_single, because the position of  $X$  (the original images in this case) is always in the center and independent of  $U$ . However, while being able to estimate rotation angles accurately in the training set, CNN\_single completely fails in the test set, because the estimation of angles relies highly on the pattern of images, which is determined by  $U$ . This also offers insight into numerical regression tasks in contemporary computer vision studies, such as object pose estimation, for which given only the images after transformation for training, a good generalization performance cannot be expected.

Secondly, for CNN backbones, computation based on image-level concatenation (instead of feature-level) is beneficial for making more accurate estimations. Fig. 6 shows that Siamese networks underperform CNN\_pair in learning all mechanisms. Much information about transformations is lost through the convolutional and max pooling operations, while more information can be preserved from the beginning with the CNN\_pair.

Additionally, we speculate that the inductive bias of CNNs fundamentally affects the effectiveness of knowledge learning. This is based on the observation of the learning curves of the three mechanisms (in Fig. A.13 in Appendix A). Fast learning on translation and scaling and a slow one on rotation can be noticed for all models, indicating that CNN models have greater difficulty learning the mechanism of rotation. Another interesting property of CNN\_pair and Siamese networks can be found (only) in learning translations. Given two images  $\mathbf{x}$  and  $\mathbf{x}_T$  both with a small square in the center, and the target value of translation  $\theta_T$ , we can obtain a (coarse) translated version of  $\mathbf{x}_T$  by optimizing  $\mathbf{x}_T$  with gradient decent according to:

$$\mathbf{x}_T \leftarrow \mathbf{x}_T - \alpha \nabla_{\mathbf{x}_T} L_{MSE}(E(\mathbf{x}, \mathbf{x}_T), \theta_T), \quad (8)$$

where  $\alpha$  is the learning rate. As shown in Fig. 7, this operation can be viewed as an approximation of the translation function  $f_T$ . Although this reversed generation of images is by no means accurate and only limited to very simple patterns, the phenomenon clearly shows what the models have learned.

Considering CNN’s properties of translation-equivariance, positional information can be encoded and operated with CNN at higher efficiency. An extensive investigation into other inductive biases is necessary for a more solid claim to be made in the future.

## 4.2. Can Knowledge be Leveraged?

In the previous section, it can be seen that effective learning can be achieved with CNN\_pair. The models are capable of making accurate estimations on parameters  $\theta$ , and this capability can be generalized to semantically different datasets. This indicates a certain degree of generalizability. Hence, using these models as building blocks, we construct the InterpretNet as described in Section 3.3. In this section, the classification performance is reported in Section 4.2.1, followed by the ablation study in Section 4.2.2 and discussion of the relationship with human’s visual perception in Section 4.2.3.

### 4.2.1. Classification Performance







In the experiment, classification is performed with the setting of covariate shift caused by rotation. To construct InterpretNet, the CNN\_pair models trained in Exp\_NOISE for the (individual) rotation learning and the identity function learning are exploited as the modules  $E$  and  $I$ , respectively. Only Exp\_NOISE is conducted to train  $E$  and  $I$ , and therefore we can conclude that semantic knowledge is not required in classification as long as the models have learned how to transform and compare images. The classifier  $C$  (or the basic classifier) is trained with original samples  $X^{train}$  in MNIST without any data augmentations. The length  $k$  of hypothesis  $H(\mathbf{x}_T^{test})$  is set to 5 and 10. The number of candidates  $N$  for  $E$  is set to 200 for each class. The confidence threshold of  $C$  is set to 0.9999.

The classification accuracy obtained on the MNIST test set, with or without rotations, is shown in Fig. 8. The first observation is that, in the case of rotated test set, the basic classifier has experienced nearly a 40% performance drop. However, the accuracy of InterpretNet has increased to 77% when  $k = 5$  (InterpretNet\_5) and further to 82% when  $k = 10$  (InterpretNet\_10). In InterpretNet,  $E$  and  $I$  are introduced for further interpretation when  $C$  is not very confident in its prediction, and they provide extra explanations about why the sample is classified as such and how it is rotated, by leveraging the knowledge of rotation with  $E$ . Additionally, this process does not affect the performance too much for the test set without rotation.

### 4.2.2. Ablation Study

**The classifier  $C$ .** InterpretNet\_noC is studied by removing  $C$  from InterpretNet. Due to the absence of  $C$  and thus the length of label space is unknown, the value of  $k$  is set to 10. It is found that InterpretNet\_noC outperforms the basic classifier by +13%, with a classification accuracy of 75% (in Fig. 8). It

Table 2: The training and test data used in the three groups of experiments for knowledge learning. Five example images are provided for each dataset to demonstrate the 2D transformations in each experiment. These transformations, shown from left to right, include the original image, rotation, translation, scaling and a combination of the three. To prevent potential artifacts generated during transformations, such as slanted image edges, a circular mask is applied to CIFAR-10 and Noise images.

Experiment	Training set	Test set
Exp_MNIST	MNIST (training) 	EMNIST (test, 'letter' division) 
Exp_CIFAR	CIFAR-10 (training, 9 classes) 	CIFAR-10 (training, the remaining class) 
Exp_NOISE	black/white noise 	MNIST (test) 

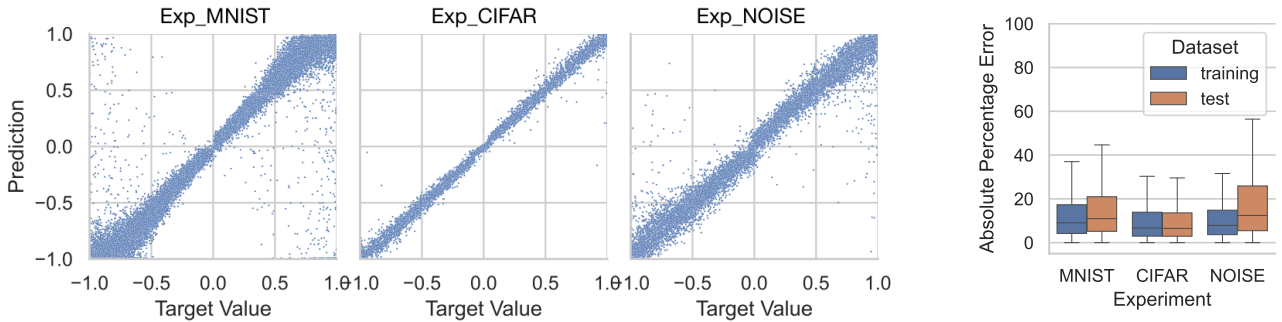


Figure 5: Performance of CNN\_pair for individual rotation learning. (left) Predictions of rotation angle vs. the ground truth (normalized to  $[-1, 1]$ ) in test set. (right) Distributions of absolute percentage errors (in %) of all data points in the dataset.

is worth noting that the performance is achieved without *any* knowledge of the handwritten digits (since both  $E$  and  $I$  are trained in Exp\_NOISE), but only through the processes of analyzing, reconstructing and matching. Furthermore, only 4% ( $200 \times 10/50000$ ) of the training data are accessed during inference. This result indicates that InterpretNet\_noC is capable of classifying characters that it does not know at all, as long as necessary references are provided, which is behaviorally similar to human beings.

**The ESTIMATOR  $E$ .** To investigate the role of  $E$  with its knowledge about rotation, an ablation study was conducted on InterpretNet\_noE by removing  $E$  from the InterpretNet. As shown in Fig. 8, the InterpretNet\_noE loses the ability to interpret rotation information and the performance on recognising rotated test set has dropped from 82% to lower than 60%. On the one hand, this indicates the importance of rotation knowledge to  $I$ , which requires instructions for reconstruction. On the other hand, since the rotated samples look very different from the candidates, it also indirectly demonstrates the effectiveness of  $I$ .

**The number of candidates.** As shown in Fig. 9, classification accuracy is greatly affected by the number of candidates. Given that  $I$  is trained on noise, the module is really sensitive to nuance differences. Therefore, to find a candidate that is very similar to a sample, a candidate pool of a proper size is re-

quired. In addition, the generation of digits can also be viewed as a mechanism. Unlike 2D transformations, the parameterization of digit generation is much more complicated [70]. While the integration of an estimation module for digit generation (as a new  $E$ ) into the existing InterpretNet would presumably reduce the required number of candidates significantly, this will, at the same time, introduce new challenges in compositionality, which involves the collaboration between multiple  $E$ s.

#### 4.2.3. Simulation of human’s visual perception

In this work, we propose InterpretNet as an exploratory simulation of human hypothesis-verification process in visual perception. Although the simulation is not reverse engineering of the human brain, based on psychological studies about cognition and behaviors, both humans and InterpretNet share similarities in how information is processed.

As human beings, we have the powerful ability to model an object with functionally easier mechanisms according to Gestalt principles [13]. This happens not only in visual perception, but also in other aspects of behaviors [71, 72], where people try to rationalize their behaviors with convincing (but sometimes incorrect) reasons. The role of  $E$  and  $I$  in InterpretNet is actually to provide explainability, with which machines can “make sense”, to some extent, of what they see. This ex-

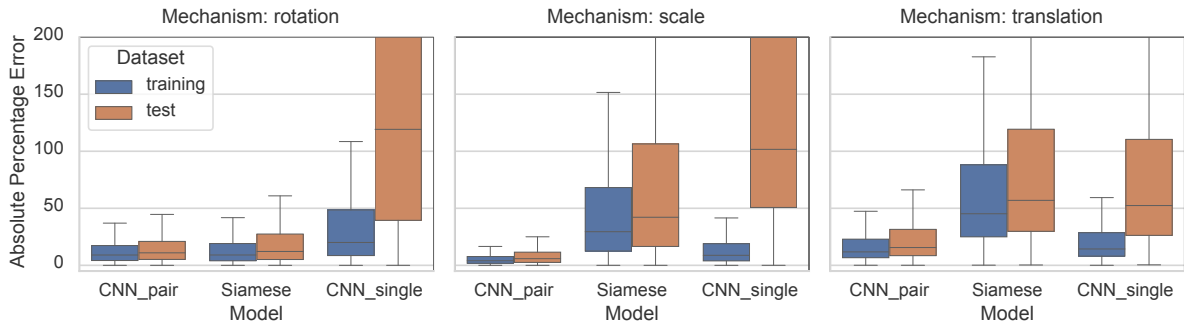


Figure 6: The performance of learning individual transformations across different models.

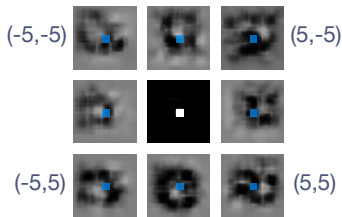


Figure 7: Images obtained with the Translation CNN\_pair through gradient descent. The image in the center is the original one  $x$ . According to the values of  $\theta$  (four of them are marked in the corners),  $x_T$  are generated through gradient descent. In each of  $x_T$ , an obvious offset of the light area from the original position (the blue dot) to the target position (the black square) can be observed.

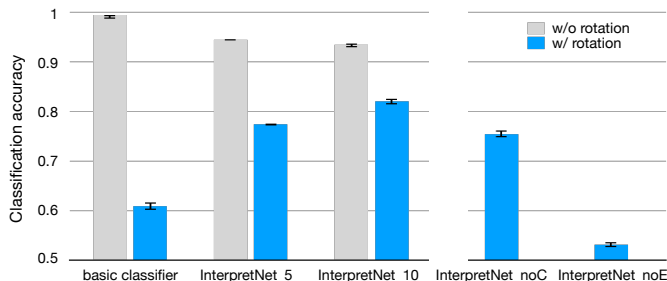


Figure 8: The performance of classification. InterpretNet\_5 and InterpretNet\_10 denote InterpretNet with hypothesis  $k = 5$  and  $k = 10$ , respectively.

plainability also provides possibilities for humans to improve the architectures, in ways that they can comprehend.

Furthermore, the simulation and imagination in brains have been studied in various works, and are proposed as the key elements in the understanding of physical scenes and counterfactual reasoning [15, 73]. Based on the model of the world in mind, humans can make predictions about the future (in a causal direction) and infer the causes of things that have happened (in an anti-causal direction). In the architecture of InterpretNet, simulations of 2D transformations in anti-causal and causal directions are enabled with the module  $E$  and the affine transformation functions, respectively, which equip the machine with an imagination space.

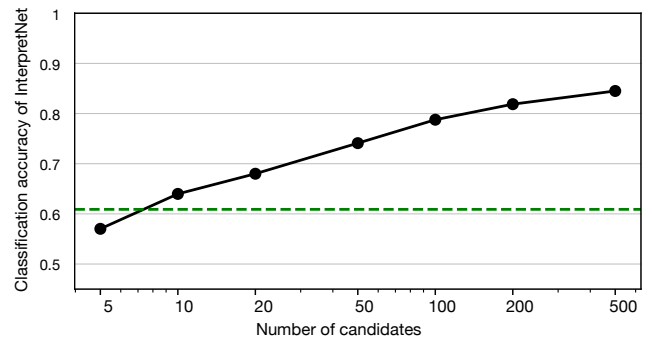


Figure 9: The classification accuracy of InterpretNet with different numbers of candidates. Performance surpasses the basic classifier (the green dash line) when  $N \geq 10$ .

## 5. Conclusion

In conclusion, this study has conducted comprehensive experiments to address the two research questions: 1) *learning* and 2) *leveraging* generalizable knowledge of 2D image transformations. Firstly, it has been demonstrated that learning generalizable knowledge of 2D image transformation mechanisms is possible if the CNN\_pair model is trained on synthetic images that are intrinsically related through the mechanism. The CNN\_pair model has exhibited significantly lower shift of mean median APE, as low as 2.5%, compared to 9.2% and 76.8% for the Siamese network and the CNN\_single, respectively. This result indicates robust generalizability of the learned knowledge, irrespective of the semantic domain of images.

Secondly, the CNN\_pair model, with its acquired knowledge, can be applied to InterpretNet and improve the performance of image classifications under covariate shift. With a single classifier, the classification accuracy drops to 60.9% (from 99.3%) after rotation. However, by leveraging the capability of transforming and comparing images, InterpretNet has improved the accuracy after rotation to 82.0% (+21.1%). The performance boost of the InterpretNet suggests the effectiveness of the simulation in human-like visual perception.

## 6. CRediT Authorship Contribution Statement

**Jiachen Kang:** Conceptualization, Methodology, Software, Investigation, Formal Analysis, Writing - original draft. **Wenjing Jia:** Conceptualization, Investigation, Validation, Formal Analysis, Writing - review & editing. **Xiangjian He:** Conceptualization, Resources, Validation, Writing - review & editing.

## 7. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## 8. Data Availability

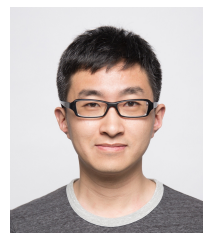
This article uses a publicly available dataset, which has been cited in the article.

## References

- [1] M. A. Alcorn, Q. Li, Z. Gong, C. Wang, L. Mai, W.-S. Ku, A. Nguyen, Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 4845–4854.
- [2] A. Barbu, D. Mayo, J. Alverio, W. Luo, C. Wang, D. Gutfreund, J. Tenenbaum, B. Katz, Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models, *Advances in neural information processing systems* 32 (2019).
- [3] I. J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, arXiv preprint arXiv:1412.6572 (2014).
- [4] J. Jo, Y. Bengio, Measuring the tendency of cnns to learn surface statistical regularities, arXiv preprint arXiv:1711.11561 (2017).
- [5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, IEEE, 2009, pp. 248–255.
- [6] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, A. Geiger, Occupancy networks: Learning 3d reconstruction in function space, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 4460–4470.
- [7] Y. Zhao, Y. Wu, C. Chen, A. Lim, On isometry robustness of deep 3d point cloud models under adversarial attacks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 1201–1210.
- [8] B. M. Lake, Compositional generalization through meta sequence-to-sequence learning, *Advances in neural information processing systems* 32 (2019).
- [9] D. Hupkes, V. Dankers, M. Mul, E. Bruni, Compositionality decomposed: How do neural networks generalise?, *Journal of Artificial Intelligence Research* 67 (2020) 757–795.
- [10] G. Bao, N. Zhuang, L. Tong, B. Yan, J. Shu, L. Wang, Y. Zeng, Z. Shen, Two-level domain adaptation neural network for eeg-based emotion recognition, *Frontiers in Human Neuroscience* 14 (2021) 605246.
- [11] Z. Wan, R. Yang, M. Huang, N. Zeng, X. Liu, A review on transfer learning in eeg signal analysis, *Neurocomputing* 421 (2021) 1–14.
- [12] L. Wang, P. Luc, Y. Wu, A. Recasens, L. Smaira, A. Brock, A. Jaegle, J.-B. Alayrac, S. Dieleman, J. Carreira, et al., Towards learning universal audio representations, in: ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 4593–4597.
- [13] K. Koffka, *Principles of Gestalt psychology*, Routledge, 2013.
- [14] E. Téglás, E. Vul, V. Girotto, M. Gonzalez, J. B. Tenenbaum, L. L. Bonatti, Pure reasoning in 12-month-old infants as probabilistic inference, *science* 332 (6033) (2011) 1054–1059.
- [15] P. W. Battaglia, J. B. Hamrick, J. B. Tenenbaum, Simulation as an engine of physical scene understanding, *Proceedings of the National Academy of Sciences* 110 (45) (2013) 18327–18332.
- [16] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, S. J. Gershman, Building machines that learn and think like people, *Behavioral and brain sciences* 40 (2017).
- [17] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, Y. Bengio, Toward causal representation learning, *Proceedings of the IEEE* 109 (5) (2021) 612–634.
- [18] G. F. Marcus, *The algebraic mind: Integrating connectionism and cognitive science*, MIT press, 2003.
- [19] S. M. Lehar, *The world in your head: A gestalt view of the mechanism of conscious experience*, Psychology Press, 2003.
- [20] A. J. Marcel, Conscious and unconscious perception: An approach to the relations between phenomenal experience and perceptual processes, *Cognitive psychology* 15 (2) (1983) 238–300.
- [21] J. A. Fodor, Z. W. Pylyshyn, Connectionism and cognitive architecture: A critical analysis, *Cognition* 28 (1-2) (1988) 3–71.
- [22] A. E. Stahl, L. Feigenson, Observing the unexpected enhances infants’ learning and exploration, *Science* 348 (6230) (2015) 91–94.
- [23] L. E. Schulz, A. Gopnik, C. Glymour, Preschool children learn about causal structure from conditional interventions, *Developmental science* 10 (3) (2007) 322–332.
- [24] C. Cook, N. D. Goodman, L. E. Schulz, Where science starts: Spontaneous experiments in preschoolers’ exploratory play, *Cognition* 120 (3) (2011) 341–349.
- [25] H. Schmidt, E. Spelke, The development of gestalt perception in infancy, *Infant Behavior and Development* 9 (1986) 329.
- [26] E. S. Spelke, Principles of object perception, *Cognitive science* 14 (1) (1990) 29–56.
- [27] Y. Bengio, T. Deleu, N. Rahaman, R. Ke, S. Lachapelle, O. Bilaniuk, A. Goyal, C. Pal, A meta-transfer objective for learning to disentangle causal mechanisms, arXiv preprint arXiv:1901.10912 (2019).
- [28] B. Recht, R. Roelofs, L. Schmidt, V. Shankar, Do imagenet classifiers generalize to imagenet?, in: International Conference on Machine Learning, PMLR, 2019, pp. 5389–5400.
- [29] X. Zhang, P. Cui, R. Xu, L. Zhou, Y. He, Z. Shen, Deep stable learning for out-of-distribution generalization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 5372–5382.
- [30] S. Chen, E. Dobriban, J. H. Lee, Invariance reduces variance: Understanding data augmentation in deep learning and beyond, *CoRR* abs/1907.10905 (2019). URL <http://arxiv.org/abs/1907.10905>
- [31] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, T. Darrell, Deep domain confusion: Maximizing for domain invariance, arXiv preprint arXiv:1412.3474 (2014).
- [32] J. Lee, Y. Lee, J. Kim, A. Kosiorek, S. Choi, Y. W. Teh, Set transformer: A framework for attention-based permutation-invariant neural networks, in: International conference on machine learning, PMLR, 2019, pp. 3744–3753.
- [33] C. Shorten, T. M. Khoshgoftaar, B. Furht, Text data augmentation for deep learning, *Journal of big Data* 8 (1) (2021) 1–34.
- [34] M. Mirza, S. Osindero, Conditional generative adversarial nets, arXiv preprint arXiv:1411.1784 (2014).
- [35] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, L. Bottou, Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion., *Journal of machine learning research* 11 (12) (2010).
- [36] E. D. Cubuk, B. Zoph, J. Shlens, Q. V. Le, Randaugment: Practical automated data augmentation with a reduced search space, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, 2020, pp. 702–703.
- [37] A. Laishram, K. Thongam, Automatic classification of oral pathologies using orthopantomogram radiography images based on convolutional neural network, *International Journal of Interactive Multimedia and Artificial Intelligence* 7 (Regular Issue) (2022) 69–77.
- [38] O. F. Kar, T. Yeo, A. Atanov, A. Zamir, 3d common corruptions and data augmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 18963–18974.
- [39] X. Hao, Y. Zhu, S. Appalaraju, A. Zhang, W. Zhang, B. Li, M. Li,



- Mixgen: A new multi-modal data augmentation, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023, pp. 379–389.
- [40] F. Muratore, T. Gruner, F. Wiese, B. Belousov, M. Gienger, J. Peters, Neural posterior domain randomization, in: Conference on Robot Learning, PMLR, 2022, pp. 1532–1542.
- [41] T. Dai, K. Arulkumaran, T. Gerbert, S. Tukra, F. Behbahani, A. A. Bharath, Analysing deep reinforcement learning agents trained with domain randomisation, *Neurocomputing* 493 (2022) 143–165.
- [42] M. Jaderberg, K. Simonyan, A. Zisserman, et al., Spatial transformer networks, *Advances in neural information processing systems* 28 (2015) 2017–2025.
- [43] X. Pan, Z. Xia, S. Song, L. E. Li, G. Huang, 3d object detection with pointformer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, 2021, pp. 7463–7472.
- [44] M. Adimoolam, S. Mohan, G. Srivastava, et al., A novel technique to detect and track multiple objects in dynamic video surveillance systems, *International Journal of Interactive Multimedia and Artificial Intelligence* 7 (Regular Issue) (2022) 112–120.
- [45] Y. Zhang, C. Wang, X. Wang, W. Liu, W. Zeng, Voxeltrack: Multi-person 3d human pose estimation and tracking in the wild, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45 (2) (2022) 2613–2626.
- [46] V. N. Nguyen, Y. Hu, Y. Xiao, M. Salzmann, V. Lepetit, Templates for 3d object pose estimation revisited: Generalization to new objects and robustness to occlusions, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 6771–6780.
- [47] L. Zhang, G.-J. Qi, L. Wang, J. Luo, Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2547–2555.
- [48] X. Wang, D. Kihara, J. Luo, G.-J. Qi, Enaet: Self-trained ensemble auto-encoding transformations for semi-supervised learning, arXiv preprint arXiv:1911.09265 2 (2019).
- [49] F. Locatello, B. Poole, G. Rätsch, B. Schölkopf, O. Bachem, M. Tschanen, Weakly-supervised disentanglement without compromises, in: International Conference on Machine Learning, PMLR, 2020, pp. 6348–6359.
- [50] J. A. Weyn, D. R. Durran, R. Caruana, Improving data-driven global weather prediction using deep convolutional neural networks on a cubed sphere, *Journal of Advances in Modeling Earth Systems* 12 (9) (2020) e2020MS002109.
- [51] K. K. Verma, B. M. Singh, Deep multi-model fusion for human activity recognition using evolutionary algorithms, *International Journal of Interactive Multimedia and Artificial Intelligence* 7 (2) (2021) 44–58.
- [52] N. Dua, S. N. Singh, V. B. Semwal, Multi-input cnn-gru based human activity recognition using wearable sensors, *Computing* 103 (7) (2021) 1461–1478.
- [53] E. Q. Wu, P. Xiong, Z.-R. Tang, G.-J. Li, A. Song, L.-M. Zhu, Detecting dynamic behavior of brain fatigue through 3-d-cnn-lstm, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 52 (1) (2021) 90–100.
- [54] M. Sameer, B. Gupta, Cnn based framework for detection of epileptic seizures, *Multimedia Tools and Applications* 81 (12) (2022) 17057–17070.
- [55] S. T. Aung, M. Hassan, M. Brady, Z. I. Mannan, S. Azam, A. Karim, S. Zaman, Y. Wongsawat, et al., Entropy-based emotion recognition from multichannel eeg signals using artificial neural network, *Computational Intelligence and Neuroscience* 2022 (2022).
- [56] A. Deng, B. Hooi, Graph neural network-based anomaly detection in multivariate time series, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 4027–4035.
- [57] M. N. Dar, M. U. Akram, R. Yuvaraj, S. G. Khawaja, M. Murugappan, Eeg-based emotion charting for parkinson’s disease patients using convolutional recurrent neural networks and cross dataset learning, *Computers in Biology and Medicine* 144 (2022) 105327.
- [58] X. Li, W. Zheng, Y. Zong, H. Chang, C. Lu, Attention-based spatio-temporal graph lstm for eeg emotion recognition, in: 2021 International Joint Conference on Neural Networks (IJCNN), IEEE, 2021, pp. 1–8.
- [59] G. Altan, A. Yayık, Y. Kutlu, Deep learning with convnet predicts imagery tasks through eeg, *Neural Processing Letters* 53 (4) (2021) 2917–2932.
- [60] K. Ellis, C. Wong, M. Nye, M. Sablé-Meyer, L. Morales, L. Hewitt, L. Cary, A. Solar-Lezama, J. B. Tenenbaum, Dreamcoder: Bootstrapping inductive program synthesis with wake-sleep library learning, in: Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation, Association for Computing Machinery, New York, NY, USA, 2021, p. 835–850.
- [61] W. Lee, H. Cho, Inductive synthesis of structurally recursive functional programs from non-recursive expressions, *Proceedings of the ACM on Programming Languages* 7 (POPL) (2023) 2048–2078.
- [62] X. Duan, X. Wang, Z. Zhang, W. Zhu, Parametric visual program induction with function modularization, in: International Conference on Machine Learning, PMLR, 2022, pp. 5643–5658.
- [63] S. Kumar, C. G. Correa, I. Dasgupta, R. Marjeh, M. Y. Hu, R. Hawkins, J. D. Cohen, K. Narasimhan, T. Griffiths, et al., Using natural language and program abstractions to instill human inductive biases in machines, *Advances in Neural Information Processing Systems* 35 (2022) 167–180.
- [64] J. Platt, Using analytic qp and sparseness to speed training of support vector machines, *Advances in neural information processing systems* 11 (1998).
- [65] S. Sabour, N. Frosst, G. E. Hinton, Dynamic routing between capsules, *Advances in neural information processing systems* 30 (2017).
- [66] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proceedings of the IEEE* 86 (11) (1998) 2278–2324.
- [67] S. Chopra, R. Hadsell, Y. LeCun, Learning a similarity metric discriminatively, with application to face verification, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), Vol. 1, IEEE, 2005, pp. 539–546.
- [68] G. Cohen, S. Afshar, J. Tapson, A. Van Schaik, Emnist: Extending mnist to handwritten letters, in: 2017 International Joint Conference on Neural Networks (IJCNN), IEEE, 2017, pp. 2921–2926.
- [69] A. Krizhevsky, G. Hinton, et al., Learning multiple layers of features from tiny images, Tech. rep., University of Toronto (2009).
- [70] B. M. Lake, R. Salakhutdinov, J. B. Tenenbaum, Human-level concept learning through probabilistic program induction, *Science* 350 (6266) (2015) 1332–1338.
- [71] M. S. Gazzaniga, The split brain revisited, *Scientific American* 279 (1) (1998) 50–55.
- [72] R. E. Nisbett, T. D. Wilson, Telling more than we can know: Verbal reports on mental processes., *Psychological review* 84 (3) (1977) 231.
- [73] J. Pearl, D. Mackenzie, *The book of why: the new science of cause and effect*, Basic books, 2018.
- [74] S. Madan, T. Henry, J. Dozier, H. Ho, N. Bhandari, T. Sasaki, F. Durand, H. Pfister, X. Boix, On the capability of neural networks to generalize to unseen category-pose combinations, Tech. rep., Center for Brains, Minds and Machines (CBMM) (2020).



**Jiachen Kang** received his BEng degree from Tsinghua University in 2006. He is currently a PhD student in the Faculty of Engineering and Information Technology, University of Technology Sydney (UTS). His research focuses on knowledge acquisition in computer vision.



**Wenjing Jia** received her PhD degree in Computing Sciences from the University of Technology Sydney in 2007. She is currently an Associate Professor at the Faculty of Engineering and Information Technology (FEIT), University of Technology Sydney (UTS). Her research falls in the fields of image processing and analysis, computer vision and pattern recognition.



**Xiangjian He** received his PhD degree from the University of Technology Sydney, Australia, in 1999. He is currently with the University of Nottingham Ningbo China and is the Director of the Computer Vision and Intelligent Perception Lab. His research interests include image processing, network security, pattern recognition, computer vision and machine learning.



## Appendix A. Additional Results

*Individual learning.* Additional results of performance of CNN\_pair for individual 2D transformation learning is shown in Fig. A.10. Similar to the result in Fig. 5, several observations for individual learning are listed as follows.

- Majority of absolute percentage errors (APE) can be controlled to below 20% for individual learning, which indicates the effectiveness of 2D transformation learning.
- There are only minor differences in the distributions of APE between the training and test sets for individual learning across all experiments, which suggests strong generalizability.

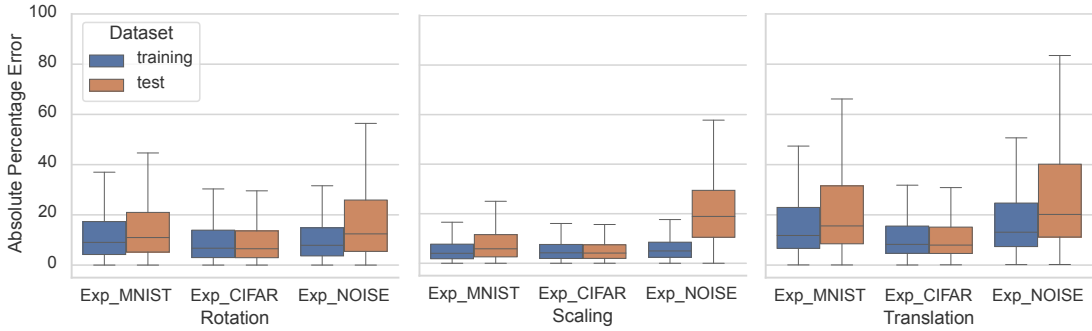


Figure A.10: Performance of CNN\_pair for individual 2D transformation learning. **(left)** Rotation. **(center)** Scaling. **(right)** Translation.

*Joint learning.* For joint learning of 2D transformation, obvious performance drop in both the training and test set can be observed in Fig. A.11, compared with the individual learning, even if the number of parameters of CNN\_pair is four times that of models for individual learning. Similar results are reported in study [74], where more accurate estimations of variables are made by separately trained models, because of the improved “selectivity and invariance at the individual neuronal level”.

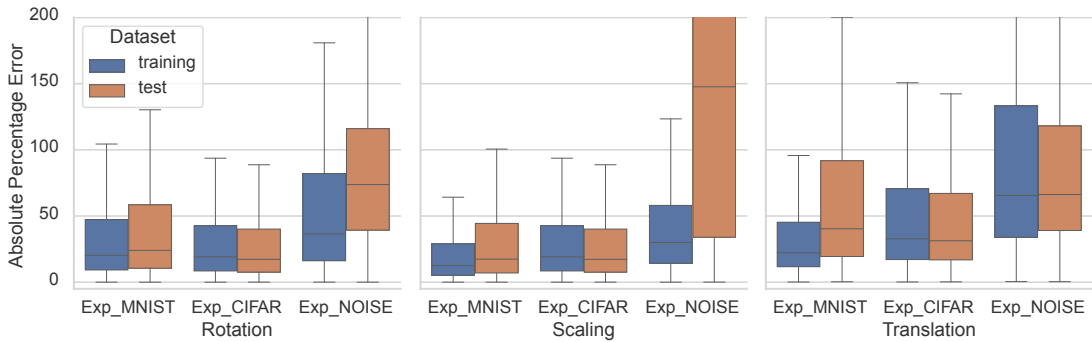


Figure A.11: Performance of CNN\_pair for joint 2D transformation learning. **(left)** Rotation. **(center)** Scaling. **(right)** Translation.

*CNN\_pair trained in Exp\_NOISE.* Although CNN\_pair exhibits strong generalization, the performance decreases to some extent when the difference between the training and test sets becomes considerably big. For instance, a larger performance gap between the training and test set in Exp\_NOISE can be noticed, compared with the other two experiments in Fig. A.10 and A.11. The most apparent characteristic in this experiment is the pattern difference between noises and hand-written digits, which implies the potential difference in exploitation of patterns during learning.

To prove this, an ablation study was conducted by altering the ratio of black to white pixels of the training data in Exp\_NOISE. As shown in Fig. A.12, the best-performing model for rotation learning is trained on 7 : 3 black/white noises. However, if the pixel values in MNIST are swapped ( *i.e.* black digits on white background), the best performance can be achieved around 4 : 6. Different ratios will provide different patterns that can be exploited in learning. The best ratio for individual learning of translation and rotation is around 7 : 3, while for scaling it is around 3 : 7, which can also explain the poor *o.o.d.* generalization performance of joint learning in Exp\_NOISE, since it is impossible for the model to learn the three transformations equally well with only one ratio.

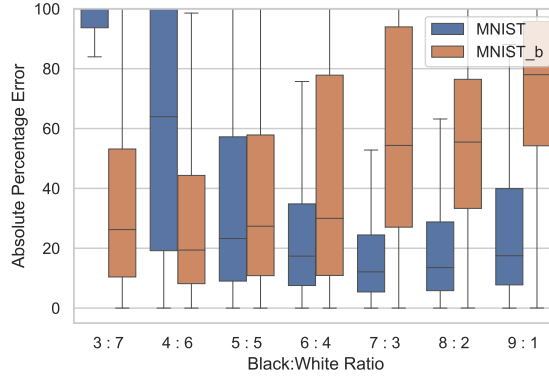


Figure A.12: Performance of CNN\_pair in rotation learning with controlled black/white pixel ratios in EXP\_NOISE. Pixel values are swapped in MNIST\_b.

*Learning curves in 2D transformation learning.* The learning curves in 2D transformation learning are shown in Fig. A.13. For all three models, fast learning on translation and scaling and a slow one on rotation can be observed.

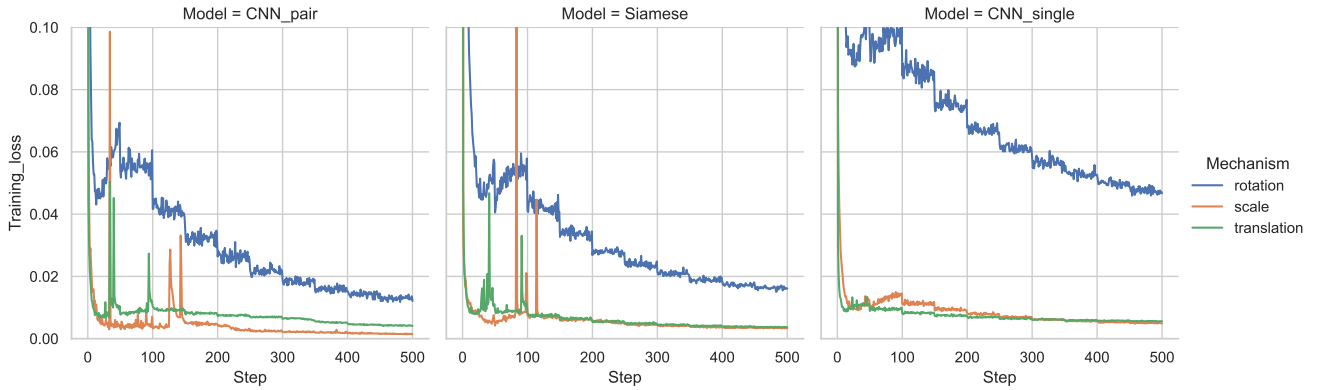


Figure A.13: The learning curves in transformation learning across different models. Fast learning on translation and scaling and a slow one on rotation can be observed for all models.

## Appendix B. Model Architecture Details

We follow the implementation in [47] to construct the three models (in Fig. 4) for knowledge learning experiments. The architectures for individual mechanism learning are shown in Table B.3. The models for joint learning are only different in channel sizes, which are all doubled in Exp\_MNIST and Exp\_NOISE, and 50% larger in Exp\_CIFAR.

Table B.3: Architecture of models for knowledge learning.

Models in Exp_MNIST and Exp_NOISE	Models in Exp_CIFAR
5×5 Conv 96, BatchNorm, ReLU	5×5 Conv 192, BatchNorm, ReLU
1×1 Conv 64, BatchNorm, ReLU	1×1 Conv 128, BatchNorm, ReLU
1×1 Conv 32, BatchNorm, ReLU	1×1 Conv 64, BatchNorm, ReLU
3×3 MaxPooling stride 2	3×3 MaxPooling stride 2
3×3 Conv 32, BatchNorm, ReLU	3×3 Conv 128, BatchNorm, ReLU
1×1 Conv 32, BatchNorm, ReLU	1×1 Conv 128, BatchNorm, ReLU
1×1 Conv 32, BatchNorm, ReLU	1×1 Conv 128, BatchNorm, ReLU
3×3 MaxPooling stride 2	3×3 MaxPooling stride 2
3×3 Conv 32, BatchNorm, ReLU	3×3 Conv 128, BatchNorm, ReLU
1×1 Conv 32, BatchNorm, ReLU	1×1 Conv 128, BatchNorm, ReLU
1×1 Conv 32, BatchNorm, ReLU	1×1 Conv 128, BatchNorm, ReLU
3×3 MaxPooling stride 2	3×3 MaxPooling stride 2
2×2 Conv 32, BatchNorm, ReLU	2×2 Conv 128, BatchNorm, ReLU
1×1 Conv 32, BatchNorm, ReLU	1×1 Conv 128, BatchNorm, ReLU
1×1 Conv 32, BatchNorm, ReLU	1×1 Conv 128, BatchNorm, ReLU
3×3 MaxPooling stride 2	3×3 MaxPooling stride 2
FC	FC
FC (Siamese networks only)	FC (Siamese networks only)