Variational Prototyping-Encoder: One-Shot Learning with Prototypical Images

Junsik Kim   Tae-Hyun Oh[†]   Seokju Lee   Fei Pan   In So Kweon

Dept. of Electrical Engineering, KAIST, Daejeon, Korea

[†]MIT CSAIL, Cambridge, US

Abstract

In daily life, graphic symbols, such as traffic signs and brand logos, are ubiquitously utilized around us due to its intuitive expression beyond language boundary. We tackle an open-set graphic symbol recognition problem by one-shot classification with prototypical images as a single training example for each novel class. We take an approach to learn a generalizable embedding space for novel tasks. We propose a new approach called variational prototyping-encoder (VPE) that learns the image translation task from real-world input images to their corresponding prototypical images as a meta-task. As a result, VPE learns image similarity as well as prototypical concepts which differs from widely used metric learning based approaches. Our experiments with diverse datasets demonstrate that the proposed VPE performs favorably against competing metric learning based one-shot methods. Also, our qualitative analyses show that our meta-task induces an effective embedding space suitable for unseen data representation.

1. Introduction

A meaningful graphic symbol visually and compactly expresses semantic information. Such graphic symbols are called ideogram,[1] which are designed to encode signal or identity information in an abstract form. They effectively convey the gist of intended signals while capturing the attention of the reader in a way that allows the reader to grasp the ideas readily and rapidly [2]. Its instant (immediate) recognition characteristic is leveraged for safety signals (e.g., traffic signs) and for better visibility and identity of commercial logos. Moreover, the compactness of iconic representativeness enables emoticons and visual hashtags [3]. Ideograms are often independent of any particular language and are comprehensible only by those with familiarity with prior conventions beyond language boundaries, *e.g.*, pictorial resemblance to a physical object.

---

[1]This is also formally called as a pictogram, pictogramme, pictograph, simply picto or icon. In this work, we interchangeably refer to an ideogram using the word "symbol" for simplicity.

Figure 1. Prototypes of symbolic icons. The top and bottom rows show traffic signs and logo prototypes, respectively.

While such symbols utilize human-perception-friendly designs, machine-based understanding of the abstract visual imagery is not necessarily straightforward due to several challenges. Original symbols in a canonical domain as shown in Fig. 1, referred to as a prototype, are rendered in a physical form by printing or displaying. These prototypes go through geometric and photometric perturbations via printing and imaging pipelines. The discrepancy between real and canonical domains introduces a large perceptual gap in the visual domain (termed *domain discrepancy*). This gap is significant in that it is difficult to close it due to extreme data imbalance between real images and a single prototype of a symbol (called an *intra-class data imbalance*). Moreover, even for real images, the annotation is typically expensive when constructing a large-scale real dataset. Although there are a few datasets with a limited number of classes, they have a noticeable class imbalance (called an *inter-class data imbalance*). Thereby, the absence of a large number of training examples for a class often raises an issue when training a large capacity learner, *i.e.*, deep neural networks.

To deal with such challenges, in this work we present a deep neural network called variational prototyping-encoder (VPE) for one-shot classification of graphic symbols. Given a single prototype of each symbol class (called a support set), VPE classifies a query into its corresponding category without requiring a large fully supervised dataset, *i.e.*, one-shot classification. The key ideas when attempting to alleviate the domain discrepancy and data imbalance issues are as follows: 1) VPE exploits existing pairs of prototypes and their corresponding real images to learn a generalizable la-
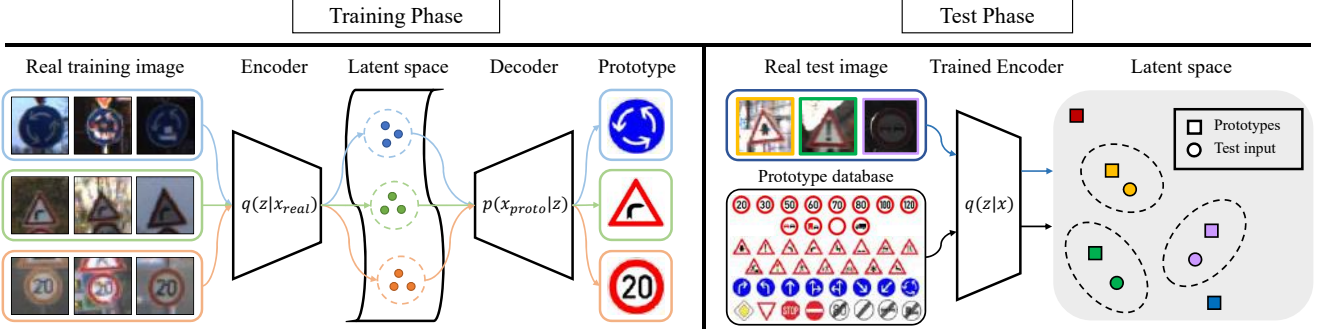
Figure 2. Illustration of the training and test phases of the variational prototyping-encoder. During training, the encoder encodes real domain input images to latent distribution $q(z|x)$. The decoder then reconstructs the encoded distribution back to a prototype that corresponds to the input image. In the test phase, the trained encoder is used as a feature extractor. Test images and prototypes in the database are encoded into the latent space. We then perform nearest neighbor classification to classify the test images. Note that classes of the prototypes in the test phase database are not used in the training phase, *i.e.*, novel classes.

tent space for unseen class data. 2) Instead of introducing a pre-determined metric, VPE learns an image translation [8] but from real images to prototype images, whereby the prototype is used as a strong supervision signal with high level visual appearance knowledge. 3) VPE leverages a variational autoencoder (VAE) [14] structure to induce a latent feature space implicitly, where the features from real data form a compact cluster around a feature point of the corresponding prototype. This is illustrated in Fig. 2.

In the test phase, as was typically done in prior works [15, 12, 31, 26], we can easily classify queries by means of a simple nearest neighbor (NN) classification scheme in the learned latent space, where the distances between a real image feature and the given prototype features are measured and the class closest to the input feature is assigned. For test purposes, we evaluate the prototypes from unseen categories in the test phase. Our method can also be used for open set classification, as an unlimited number of prototypical classes can be dealt with by regarding prototypes as an open set database.

Through empirical experimental assessments of various one-shot evaluation scenarios, we show that the proposed model performs favorably against recent metric-based one-shot learners. The improvement on traffic sign datasets is noticeably significant compared to the second best method (53.30%→83.79% on the GTSRB scenario and 58.75%→71.80% on the GTSRB→TT100K scenario) as well as on logo datasets (40.95%→53.53% on the Belga→Flickr32 scenario and 36.62%→57.75% on the Belga→Toplogos scenario). We also provide a visual understanding of VPE's embedding space by plotting t-SNE feature distributions and the average images of top-K retrieved images. The source code is publicly available. [2]

2. Related Work

In the one-shot learning context, the pioneering works of Fei-Fei *et al.* [19] hypothesize that efficiency of learning in humans may come from the advantage of prior experience. To mimic this property, they explored a Bayesian framework to learn generic *prior* knowledge from unrelated tasks, which can be quickly adapted to new tasks with few examples and forms the *posterior*. More recently, Lake *et al.* [16] developed a method of learning the concepts of the generative process with simple examples by means of hierarchical Bayesian program learning, where the learned concepts are also readily generalizable to novel cases, even with a single example. Despite the success of recent end-to-end deep neural networks (DNN) in other learning tasks, one-shot learning remains a persistently challenging problem, and hand-designed systems often outperform DNN based methods [16].

Nonetheless, in one-shot learning (including few-shot learning), the efforts to exploit the benefit of DNN is under progression. One-shot learning regime is inherently harsh due to the over-fitting issue caused by a low number of data. Thus, recent DNN based approaches have mainly been progressed either to achieve generalizable metric space with regard to unrelated task data (*i.e.*, embedding space learning) or to learn high-level strategies (*i.e.*, meta-learning).

Our method is close to the former category. Once a metric is given, non-parametric models such as the nearest neighbor (NN) enable unseen examples to be assimilated instantly without re-training; hence, novel category classification can be done by a simple NN. The following works are related: metric learning by Siamese networks [15], Quadruplet networks [12] and N-way metric learning [31, 26]. Given a metric (*e.g.*, Euclidean distance [15, 12, 26], cosine distance [31]), these approaches learn an embedding space (latent space) in the hope of generalization to novel but related domain data. Our method is different in that we do not spec-

ify a metric directly but implicitly learn an embedding space by a meta-task, *i.e.*, image translation from a real domain image to a prototype image.

Recent meta-learning approaches have been applied to few-shot learning. Santoro *et al.* [25] and Mishra *et al.* [21] take a sequence learning approach as a meta-learner so that given a series of input sequences, the learner learns high-level strategies by which possibly to solve new tasks. Ravi & Larochelle [23] and Finn *et al.* [5] seek to learn a representation that can be easily fine-tuned to new data with a few steps of gradient descent updates. Given that most meta-learner based methods [25, 21, 23, 5, 31, 26] learn high-level strategies, they typically adopt episodic training schemes that must be well-coordinated. This is contrary to the aforementioned metric learning based approaches [15, 12] including our method, where the training steps are usually rather straightforward.

The methods discussed above focus on cases in which examples in supported set and a query are from the same domain. In our problem setup, the significant discrepancy between real-world query images and the prototype in the support set introduces new challenges. There have been few attempts related to one-shot learning with prototypes. Jetley *et al.* [10] proposed a feature transform approach to align features of real images with pre-defined hand-crafted features of prototypes. Kim *et al.* [12] is the work closest to our method. They proposed the learning of co-domain embedding using deep quadruplet networks in an end-to-end manner so that an embedding of prototype and real-world images are mapped into a common feature space. Recently, Snell *et al.* [26] proposed prototypical networks for few-shot learning in an extension of Vinyals *et al.* [31]. However, their definition of a prototype differs from ours in that their prototype is defined according to the mean centroid of a class on the same domain with queries, while our prototype is a prototypical image.

3. Proposed Method

We use a one-shot learning approach simliar to metric learning based methods [15, 12, 26, 31], which learn an embedding space as general as possible by means of a metric comparison. Such approaches consist of two steps: 1) a training step to learn the embedding space with massive data (generic prior knowledge), and 2) a test step involving NN classification with embeddings of novel class data and their support set. This approach assumes that the data used in the training step is unrelated to the class of the test phase but has a distribution similar to that of the test data. Moreover, the embedding is expected to be informative so that one to five support samples (one-shot to few-shot) for each novel class can be sufficiently generalized.

The Variational prototyping-encoder (VPE) differs from metric learning in terms of how it induces a generalized embedding space. Instead of determining a user-selected metric to induce an embedding space, VPE learns a generative model with a continuous distribution of data. VPE seeks the embedding space via a meta-task; conditional image translation from a real image to a prototype. Additionally, VPE guides distribution learning using prior information about prototypes.

In this paper, we denote a scenario with a support set consisting of C classes with K samples per class as C–way K–shot classification. We assume that a single prototype (K=1) is given for each class as a supported sample, *i.e.*, one-shot classification with a single prototype.

3.1. Variational Prototyping-Encoder

Let us consider a paired dataset $\mathcal{X} = \{(\mathbf{x}, \mathbf{t})^{(i)}\}_{i=1}^{N}$, where $\mathbf{x}$ is the real image sample, $\mathbf{t}$ denotes its corresponding prototype image, and we assume respective i.i.d. samples. In our scenario, each class has only a single prototype $\mathbf{t}$ which acts as a label. We assume a data generation process similar to a variational autoencoder (VAE) [14], but the generated target value is not data $\mathbf{x}$ but $\mathbf{t}$: *i.e.*, a latent code $\mathbf{z}^{(i)}$ is generated from a prior distribution $p_\theta(\mathbf{z})$, after which a prototype $\mathbf{t}^{(i)}$ is generated from a conditional distribution $p_\theta(\mathbf{x}|\mathbf{z})$. Because this process is hidden, the parameter θ and the latent variables $\mathbf{z}^{(i)}$ are unknown. Thus, we approximate the inference by means of a variational Bayes method.

The parameter approximation is done via marginal likelihood maximization. Each log marginal likelihood of the individual prototype $\log p_\theta(\mathbf{t}^{(i)})$ can be lower bounded by

$$
\begin{aligned}
\log p_\theta(\mathbf{t}) &= \log \int_{\mathbf{z}} p_\theta(\mathbf{t}, \mathbf{z}) = \log \int_{\mathbf{z}} p(\mathbf{t}, \mathbf{z}) \tfrac{q_\phi(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{x})} \\
&= \log \left(\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \tfrac{p(\mathbf{t}, \mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})} \right) \\
&\geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log p_\theta(\mathbf{t}, \mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x}) \right] \\
&\quad \text{(by Jensen's inequality)} \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log p_\theta(\mathbf{t}|\mathbf{z}) \right] - D_{KL} \left[q_\phi(\mathbf{z}|\mathbf{x}) || p_\theta(\mathbf{z}) \right],
\end{aligned}
\tag{1}
$$

where $D_{KL}[\cdot]$ is the Kullback-Leibler (KL) divergence, and a proposal distribution $q_\phi(\mathbf{z}|\mathbf{x})$ is introduced to approximate the intractable true posterior. The distributions $q_\phi(\mathbf{z}|\mathbf{x})$ and $p_\theta(\mathbf{t}|\mathbf{z})$ are termed a probabilistic encoder and decoder (or a recognition model and a generative model) respectively. By maximizing the variational lower bound in Eq. (1), we can determine the model parameters ϕ and θ of the encoder and decoder.

Eq. (1) is different from the VAE [14]. The VAE is derived from the marginal likelihood over the input data $\mathbf{x}$, and its lower bound models the self-expression of the input, as

$$
\log p_\theta(\mathbf{x}) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log p_\theta(\mathbf{x}|\mathbf{z}) \right] - D_{KL} \left[q_\phi(\mathbf{z}|\mathbf{x}) || p_\theta(\mathbf{z}) \right].
\tag{2}
$$

In this formulation, $\mathbf{x}$ is encoded to $\mathbf{z}$ and reconstructed from $\mathbf{z}$, while our method encode the input $\mathbf{x}$ to $\mathbf{z}$ and translate

to a prototype **t** like image-to-image translation [8]. Since prototypes are on a canonical domain with canonical color without perturbation in real objects, our method translates real image inputs to the corresponding prototypical images invariant to real-world perturbations such as background clutter, geometric and photometric perturbations. In this sense, VPE is related to the denoising autoencoder [30, 1] in that VPE acts as a real-world perturbation normalization and may result in embeddings (latent **z**) invariant or robust to the perturbations.

In order to efficiently train the parameters by stochastic gradient descent (SGD), we follow Kingma and Welling [14] to derive a differentiable surrogate objective function by assuming Gaussian latent variables and drawing samples $\{\mathbf{z}^{(s)}\}_{s=1}^S$ from $q_\phi(\mathbf{z}|\mathbf{x})$. The empirical loss is then derived as follows:

$$\mathcal{L}(\mathbf{x}, \mathbf{t}; \theta, \phi) = \frac{1}{S}\sum_{s=1}^S -\log p_\theta(\mathbf{t}|\mathbf{z}^{(s)}) + D_{KL}[q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z})].$$

$$(3)$$

The reparameterization trick [14] is used for Eq. (3) to be differentiable, whereby $q_\phi(\mathbf{z}|\mathbf{x})$ is re-parameterized with a neural network $g_\phi(\cdot)$, *i.e.*, $\mathbf{z}^{(s)}$ is sampled by $\mathbf{z}^{(s)} = g_\phi(\mathbf{x}^{(i)}, \boldsymbol{\epsilon}^{(s)}) = \boldsymbol{\mu}^{(i)} + \boldsymbol{\sigma}^{(i)} \odot \boldsymbol{\epsilon}^{(s)}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\odot$ denotes element-wise multiplication. In addition, the decoder $p_\theta(\mathbf{t}|\mathbf{z})$ is modeled by a neural network. We can efficiently minimize Eq. (3) by SGD with a mini-batch.

In Eq. (3), the first and second term correspond to the reconstruction error and distribution regularization term respectively. KL divergence regularizes the latent space by encouraging the distribution of **z** follows the prior distribution, which prevents the distribution from collapsing while mapping similar data inputs to nearby locations in the latent space. Furthermore, the loss induces the mapping of various real images to a single prototype image of the same class. This enables the distribution of the latent vectors of real images within the same class to be encapsulated by conditioning its prototype.

For the reconstruction loss in Eq. (3), any reconstruction loss can be used, from basic losses (ℓ_1- and ℓ_2-norm) to advanced losses (perceptual loss [7] and generative adversarial loss [6, 17]). We used the simple binary cross entropy (BCE) loss with real valued targets in $[0, 1]$, finding that it is sufficiently efficient for prototypes because many prototypes consist of primary colors within the range of $[0, 1]$. More exploration of loss functions will lead to improvement.

Test phase. The learned encoder is only used as a feature extractor. Given a novel class support set of prototypes, we initially extract their features from the encoder and store them in the support set, (one-shot learning). Subsequently, when an input query is given, we extract its feature by the encoder and classify by NN classification by retrieving the support set (Fig. 2). Because we assume Gaussian latent

variables, we can measure the similarity by Euclidean or Mahalanobis distances. In this work, we simply use the Euclidean distance for NN classification. We leave the development of advanced metrics as a future work.

Comparison with other approaches. In classification, metric learning based one-shot methods [15, 12, 26, 31] learn non-linear mappings suitable for the given metric distances with labels. Label information groups data based on discrete decisions as to whether samples belong to the same class or not. This tends to be discriminative for the seen classes. However, it would be difficult to expect the features of images from unseen classes to be distributed meaningfully over the feature space learned in such a manner.[3] Therefore, several methods have attempted to alleviate the shortage of the metric loss, such as multiple pairwise regularization [12] and attentional kernel with conditional embedding [31], but still limited.

Without directly fixing a metric, our model learns an embedding space in a wholly different manner. VPE with the prototype reconstruction loss learns the meta-task of normalizing real images and indirectly learns the relative similarities of real images as well as latent features according to the degree of appearance similarity with the corresponding prototypes. We will show in the experimental section that learning appearance similarity in the image domain allows better generalization.

3.2. Network architecture

We build an encoder with three convolution layers followed by one fully connected layer each for mean and variance predictions. Each convolution layer has a stride size of 2, downsizing the feature map by a factor of 2. Every convolution layer is followed by batch normalization and leaky ReLU. The final layer is a fully connected layer converting a feature map into a predefined latent variable size. The convolution filter size and latent variable size follow that of the Idsia network [4] which has been the best traffic sign classification network within the GTSRB benchmark [27]. Layers of the decoder are in an inverse order of the encoder layers; *i.e.*, a fully connected layer followed by three convolution layers. We upsample by a factor of 2 before each convolution to recover the feature size to the original input size. All convolution kernels in the decoder are set to 3×3. As in the encoder, every convolution in the decoder is followed by batch normalization and leaky ReLU.

3.3. Data augmentation

We apply random rotation and horizontal flipping to both the real images and prototypes identically to train our networks. Augmentation diversifies the training samples including the prototypes. We can easily imagine that a sign

---

[3]We compare t-SNE visualizations of several metric learning approaches in the supplementary material offering support of this claim.

with the right directional arrow can become an arrow sign with any directional form after augmentation. This helps the generalization of our network, and we observed that it improves the performance noticeably, whereas it does so subtly in other metric learning methods.

4. Experiment

In this section, we first describe the data set configuration and the overall experiment setup, and then implementation details. We compare the following methods for one-shot classification and retrieval tasks: Siamese networks [15] (SiamNet), Quadruplet networks [12] (QuadNet), Matching networks [31] (MatchNet) and the proposed networks (VPE). We also present additional qualitative analyses, t-SNE visualization, a distance heat map between prototypes and real images, and prototype reconstruction.

Dataset	GTSRB	TT100k	BelgaLogos	FlickrLogos-32	TopLogo-10
Instances	51,839	11,988	9,585	3,404	848
Classes	43	36	37	32	11

Table 1. Symbol dataset specifications.

Datasets and experiment setup. The evaluation is conducted on two traffic sign datasets and three logo datasets with different training and test set selections. The size and number of classes for each dataset are described in Table 1. For detailed explanations about the datasets and more image visualizations, please refer to the supplementary material.

To validate our one-shot learning method, we perform a cross-dataset evaluation by separating the training and test datasets, which is a more challenging setup compared to the use of splits within a single dataset. We denote 'All' for evaluating the entire dataset and 'Unseen' for evaluating the dataset excluding the classes contained in a training set. The dataset on the left side of an arrow is used as a training set while that on the right side of an arrow is used as a test set (Table 2 and Table 3), *e.g.*, GTSRB→TT100k.

For logo classification, BelgaLogos [11, 18], FlirckrLogos-32 [24] and TopLogo-10 [28] are used. BelgaLogos is used as a training set and remaining datasets are used as the test and validation sets. For example, in the Belga→Flickr32 case, TopLogo-10 is used as a validation set. BelgaLogos and FlickrLogos-32 share four common classes, and BelgaLogos and Toplogo-10 share five common classes. We exclude the common classes in the "Unseen" test. For traffic sign classification, the GTSRB [27] and TT100K [33] datasets are used. For the GTSRB→TT100k scenario, we train the model on GTSRB and report the best accuracy tested on TT100K. GTSRB and TT100K shares four common classes.

While the entire dataset is used for training and testing during the cross-dataset evaluation, the GTSRB experiment is performed using only the GTSRB dataset with splits.

	GTSRB → GTSRB	GTSRB → TT100k	
Split	Unseen	All	Unseen
No. classes	21	36	32
No. support set	(22+21)-way	36-way	
SiamNet [15]	22.45	22.73	15.28
SiamNet+aug	33.62	28.36	22.74
QuadNet* [12]	45.2*	42.3*	N/A
MatchNet [31]	26.03	53.16	49.53
MatchNet+aug	53.30	62.14	58.75
VPE (48x48)	55.30	52.08	49.21
VPE+aug	69.46	66.62	63.91
VPE+aug+stn	74.69	66.88	64.07
VPE (64x64)	56.98	55.58	53.04
VPE+aug	**81.27**	**68.04**	**64.80**
VPE+aug+stn	**83.79**	**73.98**	**71.80**
VAE	20.67	33.14	29.04
VAE+aug	22.24	32.10	27.98

Table 2. One-shot classification (Top 1-NN) accuracy (%) on traffic sign datasets. The numbers marked with "*" are quoted from their papers. VPE on two different input resolutions, 48×48 and 64×64, are reported for the evaluations. The best accuracy is marked in blue, and the second best is shown in sky blue.

	Belga → Flickr32		Belga → Toplogos	
Split	All	Unseen	All	Unseen
No. classes	32	28	11	6
No. support set	32-way		11-way	
SiamNet [15]	23.25	21.37	37.37	34.92
SiamNet + aug	24.70	22.82	30.84	30.46
QuadNet [12]	40.01	37.72	39.44	36.62
QuadNet + aug	31.68	28.55	38.89	34.16
MatchNet [31]	45.53	40.95	44.35	35.24
MatchNet+aug	38.54	35.28	28.46	27.46
VPE	28.71	27.34	28.01	26.36
VPE+aug	**51.83**	**50.25**	**47.48**	**41.82**
VPE+aug+stn	**56.60**	**53.53**	**58.65**	**57.75**
VAE	25.01	25.48	21.90	15.89
VAE+aug	27.17	27.31	23.30	18.59

Table 3. One-shot classification (Top 1-NN) accuracy (%) on logo datasets. The best accuracy is marked in blue and the second best is shown in sky blue.

Among a total of 43 classes in GTSRB, we select 22 classes as seen and the remaining 21 classes as unseen. GTSRB has two data partitions: the train and test partitions. We trained a model with the training set of the 22 seen classes

Figure 3. Average image of top 100 images retrieved by querying prototypes. A clearer image represents a higher retrieval performance. The classes shown are selected from unseen classes.

and evaluate the performance on the test set of all 43 classes. The 21 unseen class samples in the training set are used for validation. This scenario is unique in that the support set contains all of the seen and unseen prototypes. Because the random chance accuracy of this case becomes far lower, this is a more difficult setup than the typical one-shot evaluation scenario, where a support set is assumed to contain only unseen samples. In this setup, we can determine whether a model is biased toward seen classes. The details of the GTSRB experiment setup follow the work of Kim *et al.* [12].

Implementation details. For a fair comparison, all of the methods in this experiment use IdsiaNet [4] as a base network. We tune to obtain the best performance of the methods, and we use the ADAM optimizer [13] with a learning rate of 10^{-4}, $\beta = (0.9, 0.999)$, $\epsilon = 10^{-8}$ and a mini-batch size of 128 to train the networks. The original implementations of SiamNet and MatchNet[4] are designed for character classification; hence, a base network change is necessary. We found that the substitution of the base networks significantly improved the performance outcomes. We use input sizes of 48×48 for traffic sign data and 64×64 for logo data but also test different resolution effects as a short ablation study, as shown in Table 2. The input dimension of the first fully connected layer is adjusted according to the input size so that the final dimension of embedding is fixed at 300 for all methods regardless of the input size. The rationale behind a larger size for logos is their various aspect ratios. We maintain the aspect ratio by resizing a larger axis of an image to fit the network input size with zero padding.

We also found that SiamNet performs very poorly when trained using prototypes as a query. Therefore, we trained SiamNet using only real images for both query and positive, negative sample pairs. QuadNet is reproduced using IdsiaNet

and is evaluated on the logo datasets. However, the original implementation fusing two Siamese networks performed poorly on logos. We modified QuadNet to share all of the parameters of the networks in order to stabilize the training instead of using two Siamese networks. We conjecture that the failure of the original implementation on logos stems from a quality of the training set. GTSRB is larger than logo datasets containing samples of a higher quality, whereas logo datasets have fewer samples, and some images are severely distorted, including non-rigid transformations, *e.g.*, logos printed on curved bottles or wrinkled clothes.

The term *aug* represents the random flip and rotation augmentation applied, and *stn* is a spatial transformer [9] attached to the encoder part, *i.e.*, the improved IdsiaNet suggested by the Moodstock team.[5] For the *stn* version, the spatial transformer modules are applied before the $1st$ and $3rd$ convolution layers in the encoder part. By doing this, we can show that the proposed method has the potential to be improved further if advanced techniques are adopted. Prototype images and real images are randomly sampled at a 1:200 ratio during training.

4.1. One-shot classification (Real to prototypes)

The one-shot classification performances are reported in Table 2 and Table 3. VPE and its variants perform better than competing approaches in most cases. The margin is significant in the traffic sign task while less of an improvement was noted on logo datasets. We surmise that this performance gap comes from the quality of the training dataset. As mentioned earlier, GTSRB is the largest dataset among the five datasets, and traffic sign images are well localized with consistent aspect ratios, whereas logos are more challenging

---

[4]MatchNet implementation are based on, https://github.com/gitabcworld/MatchingNetworks

[5]Their experiment achieved a meaningful performance improvement of IdsiaNet on the traffic sign classification. For more detail, please refer to, https://github.com/moodstocks/gtsrb.torch
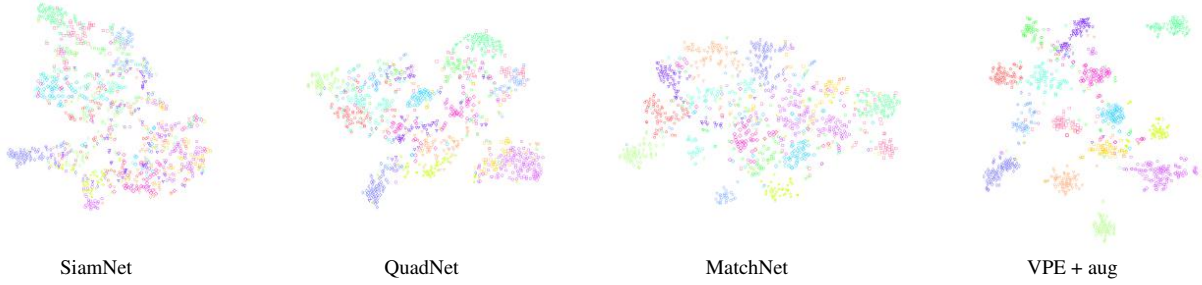
Figure 4. t-SNE visualization of features. Features are randomly sampled from 15 unseen classes of Belga→Flickr32 scenario.

due to various aspect ratios, color variations, and non-rigid deformation.

Interestingly, the augmentation improves VPE noticeably, though it has less of an effect with the other approaches. A possible explanation for this tendency is that VPE learns a pseudo image transform process and tends to measure a type of perceptual similarity which is less sensitive to subtle input changes. This would not be the case with direct metric learning methods, as subtle perceptual changes such as flipping in the input domain do not have to be mapped to similar embedding vectors. Refer to the distance heat map shown in Fig. 5

We emphasize the GTSRB scenario, of which the support set used in the test phase involves seen classes during training as well as unseen novel classes. This allows us to measure overfitting to seen classes. This is an evaluation different from typical one-shot classification setups, where a support set does not contain any samples from training classes, making the process far easier. In this scenario, MatchNet shows poor performance without augmentation. We conjecture that this is due to the attentional kernel, which is biased to favor seen classes.

The VAEs in Tables 2 and 3 are models that share the same architecture with our VPE, but trained with variational auto-encoding loss [14] without prototypes. It is reported as a reference to show how VAE performs without prototype learning. The low performance of VAE has two possible causes: 1) the lack of supervision to reduce the domain gap between the real and prototype domains, and 2) the lack of explicit information to induce clustering effects according to actual classes, which makes the VAEs difficult to adjust which level they should cluster or distinguish across samples.

4.2. Image retrieval test (Prototypes to real)

Average image [22, 32] can provide an intuitive visual understanding of multiple images. In this experiment, we summarize image retrieval results using average images. With the trained one-shot models, by querying prototypes, images are retrieved based on the metrics of each method. An average of the retrieved image qualitatively visualizes the discriminative power of the learned embeddings of the models. A fine average image is obtained only if there are negligi-

AUC	GTSRB → GTSRB	GTSRB → TT100k	Belga → Flickr32	Belga → Toplogos
SiamNet	8.75	4.83	20.56	18.13
Quadnet	n/a	n/a	32.40	20.51
MatchNet	57.99	41.00	44.47	46.13
VPE+aug	64.77	41.79	48.61	49.39
VPE+aug+stn	85.29	64.04	63.87	70.22

Table 4. AUC score of retrieval experiments.

ble outliers in the retrieved results. We provide average images by retrieval along with prototypes for comparison (Fig. 3). The result clearly shows that VPE is effective for a comparison in the opposite direction, *i.e.*, prototype → real images.

While average images provide qualitative measure of the retrieval task, we also report the quantitative retrieval performance in Table 4 using the area under the precision-recall curve (AUC). The relative retrieval performance between the competing approaches are similar to that of the one-shot experiments (Sec. 4.1).

4.3. Additional analyses

Similarity measure. One-shot classification focuses on general classification capability including that for unseen classes. Understanding image similarity and dissimilarity is an important capability for one-shot classification. Metric-based approaches adopt metric losses induced from labels, semantically coarser information without image level similarity,
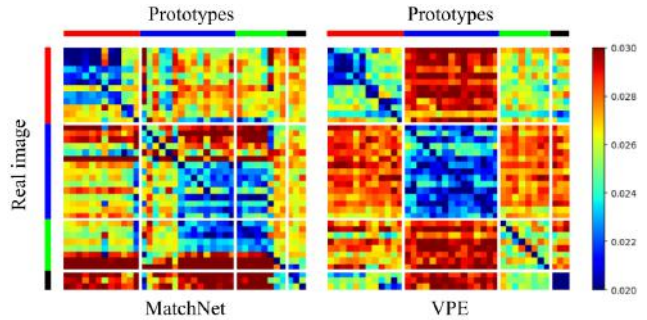

Figure 5. Average distances between real images and prototypes from GTSRB scenario are visualized as heatmap matrices.

Figure 6. The VPE output on GTSRB scenario.

while the proposed method uses appearance similarity and thus semantically finer information.

To demonstrate the quality of learned image similarity further, we show, in Fig. 5, the average distance matrix between real images and prototypes from the GTSRB dataset. Each column of distance matrices is l_1 normalized for visualization purposes. The GTSRB dataset has 38 classes that are categorized into four groups: Prohibitory, Danger, Mandatory and Others. Classes within the same category have a similar external shape while differing in terms of the interior contents. Subsequently, we mark the classes of each group with one color along the x-axis and y-axis of the matrices and use red, blue, green and black for the four groups listed above, respectively. The diagonal of the matrix represents the distance between corresponding pairs of real images and prototypes. We compare the distance matrices between MatchNet and the proposed VPE. The VPE distance matrix clearly shows a block patterned distance map, indicating that VPE captures appearance similarity in the latent space. On the other hand, although MatchNet show short distances along diagonal, there is no clear block pattern aligned with category sets.

Embedding visualization. In Fig. 4, We compare t-SNE [20] plots of the embedding spaces of the methods to understand the learned embeddings of unseen data. We assign colors according to class labels to observe the discriminative behavior. VPE shows a clear separation of sample points, whereas the competing approaches show partially mixed distributions. This distribution difference is consistent with the results from the one-shot classification experiment. It would suggest that the appearance based loss leads to better learning of the general characteristics of symbols as apposed to direct metric losses.

Prototype reconstruction. While the reconstruction task is an auxiliary task for training the proposed VPE networks, for a better understanding of the image translation behavior to unseen data, we visualize the generated outputs in Fig. 6. The model robustly generates prototypes of seen classes regardless of motion blur, illumination variations, or low resolutions. While the generation performance is not accurate for unseen classes, it still captures some level of the characteristics of these classes in the input images. It is interesting to note that VPE feasibly handles high-level categories, such as prohibitory (red circle) and danger (red triangle) categories. Although the fine-details of the symbol contents are not accurate, the locations of the blobs are roughly aligned with the contents in the prototypes. This suggests that even the rough generation is still effective for NN classification in the latent space and may apply to a high-level conceptual understanding of novel contexts.

5. Conclusion

We present a new one-shot learning approach based on a generative loss. The key idea of the proposed VPE invloves the use of reconstruction loss to learn to induce indirect perceptual similarities of real images and their corresponding prototypes, as opposed to the use of a pre-determined metric. A prototype reconstruction experiment (Fig. 6) demonstrated that our VPE implicitly learns favorable knowledge about how a real image can be neutralized against real-world perturbations, such as radiometric and geometric perturbations. VPE appears to capture high level prototype concepts from images of unseen classes distorted by real world perturbations to some extent. This is fundamentally different from metric learning approaches, as they use label information to group available data in the training phase, making it difficult to expect the generalization of similarities to unseen classes.

We quantitatively and qualitatively validated the performance of the proposed methods on multiple datasets and demonstrated its favorable performance over competing approaches. Despite the noticeable performance improvement of VPE, it is simple to train and the resulting architecture is simple as well. In this regard, the principal behind VPE would lead to various applications in the future.

References

[1] Y. Bengio, L. Yao, G. Alain, and P. Vincent. Generalized denoising auto-encoders as generative models. In *Advances in Neural Information Processing Systems*, 2013. 4

[2] M. A. Borkin, Z. Bylinskii, N. W. Kim, C. M. Bainbridge, C. S. Yeh, D. Borkin, H. Pfister, and A. Oliva. Beyond memorability: Visualization recognition and recall. *IEEE transactions on visualization and computer graphics*, 22(1):519–528, 2016. 1

[3] Z. Bylinskii, S. Alsheikh, S. Madan, A. Recasens, K. Zhong, H. Pfister, F. Durand, and A. Oliva. Understanding infographics through textual and visual tag prediction. *arXiv preprint arXiv:1709.09215*, 2017. 1

[4] D. CireşAn, U. Meier, J. Masci, and J. Schmidhuber. Multi-column deep neural network for traffic sign classification. *Neural networks*, 32:333–338, 2012. 4, 6

[5] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, 2017. 3

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014. 4

[7] X. Hou, L. Shen, K. Sun, and G. Qiu. Deep feature consistent variational autoencoder. In *IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2017. 4

[8] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 3

[9] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, 2015. 6

[10] S. Jetley, B. Romera-Paredes, S. Jayasumana, and P. Torr. Prototypical priors: From improving classification to zero-shot learning. In *British Machine Vision Conference*, 2015. 3

[11] A. Joly and O. Buisson. Logo retrieval with a contrario visual query expansion. In *Proceedings of the 17th ACM international conference on Multimedia*, 2009. 5, 11

[12] J. Kim, S. Lee, T.-H. Oh, and I. S. Kweon. Co-domain embedding using deep quadruplet networks for unseen traffic sign recognition. In *AAAI*, 2018. 2, 3, 4, 5, 6, 11, 13

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[14] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014. 2, 3, 4, 7

[15] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, 2015. 2, 3, 4, 5, 13

[16] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015. 2

[17] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. In *International Conference on Machine Learning*, 2016. 4

[18] P. Letessier, O. Buisson, and A. Joly. Scalable mining of small visual objects. In *Proceedings of the 20th ACM international conference on Multimedia*, 2012. 5, 11

[19] F.-F. Li, R. Fergus, and P. Perona. A bayesian approach to unsupervised one-shot learning of object categories. In *IEEE International Conference on Computer Vision*, 2003. 2

[20] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008. 8, 11

[21] N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel. A simple neural attentive meta-learner. In *NIPS 2017 Workshop on Meta-Learning*, 2017. 3

[22] A. Oliva and A. Torralba. The role of context in object recognition. *Trends in cognitive sciences*, 11(12):520–527, 2007. 7

[23] S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. In *International Conference on Learning Representations*, 2017. 3

[24] S. Romberg, L. G. Pueyo, R. Lienhart, and R. Van Zwol. Scalable logo recognition in real-world images. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, 2011. 5, 11

[25] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap. Meta-learning with memory-augmented neural networks. In *International Conference on Machine Learning*, 2016. 3

[26] J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, 2017. 2, 3, 4

[27] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32:323–332, 2012. 4, 5, 11

[28] H. Su, X. Zhu, and S. Gong. Deep learning logo detection with data expansion by synthesising context. In *IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2017. 5, 11

[29] A. Tüzkö, C. Herrmann, D. Manger, and J. Beyerer. Open set logo detection and retrieval. *arXiv preprint arXiv:1710.10891*, 2017. 11

[30] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(Dec):3371–3408, 2010. 4

[31] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, 2016. 2, 3, 4, 5, 13

[32] J.-Y. Zhu, Y. J. Lee, and A. A. Efros. Averageexplorer: Interactive exploration and alignment of visual data collections. *ACM Transactions on Graphics (TOG)*, 33(4):160, 2014. 7

[33] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu. Traffic-sign detection and classification in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 5, 11

6. Supplementary materials

Here, we present additional details pertaining to the datasets and experiments that could not be included in the main text due to space constraints. All figures and references in this supplementary file are self-contained.

The contents included in these supplementary materials are as follows: 1) The network architecture, 2) Detail descriptions of the datasets used, 3) Embedding space visualization, and 4) Qualitative results of image retrieval.
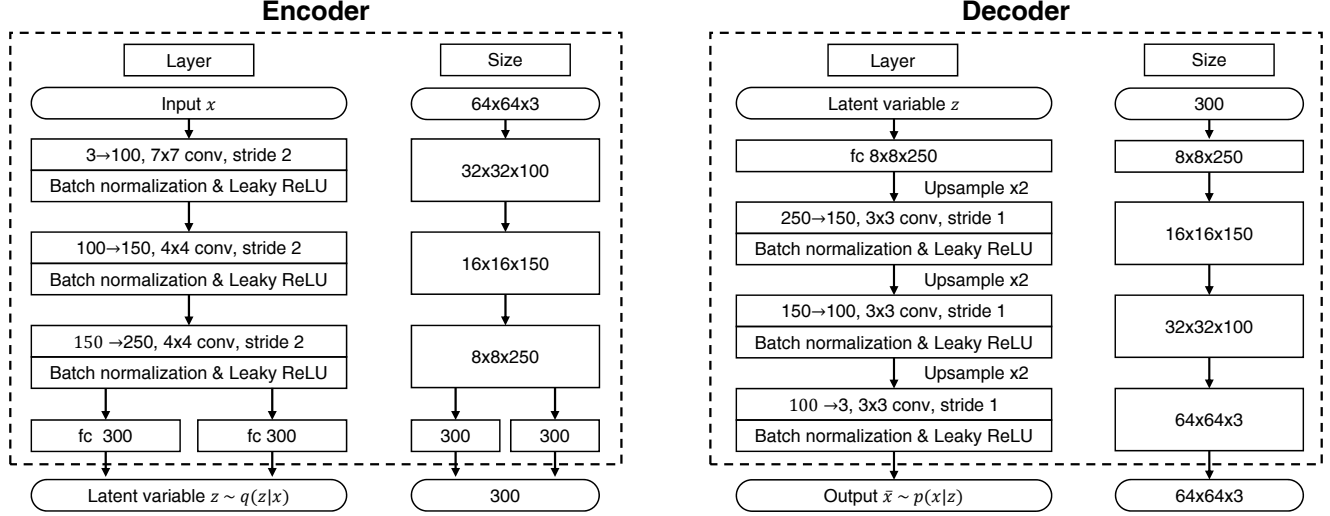
6.1. Architecture

Figure 7. Architectures specifications of encoder and decoder blocks of the proposed variational autoencoder.

The detailed VPE network architecture is shown in Fig. 7.

6.2. Datasets

Dataset	GTSRB	TT100k	BelgaLogos	FlickrLogos-32	TopLogo-10
Instances	51,839	11,988	9,585	3,404	848
Classes	43	36	37	32	11

Table 5. Symbol dataset specifications

In this section, we present the details of each dataset used for the experiments in the main text. Table 5 is provided to summarize the statistics of the datasets.

GTSRB GTSRB [27] is the largest dataset for traffic-sign recognition. It contains 43 classes categorized into three larger categories: prohibitory, danger and mandatory. The dataset contains illumination variations, blur, partial shadings and low-resolution images as well as imbalanced sample distribution. The training set contains 39,209 images and the test set contains 12,630 images.

TT100K Tsinghua-Tencent 100K (TT100K) [33] is a Chinese traffic sign detection dataset that includes more than 200 classes. We cropped traffic sign instances from scenes to build a classification dataset. We filtered out instances with side lengths of less than 20 pixels because they are either not recognizable or miss annotated. Among more the defined classes, the 36 classes are selected for the evaluation that have available corresponding prototypes and a sufficient number of samples. For more details about the TT100K dataset, please refer to the work of Kim *et al.*[12].

FlickrLogos-32 Dataset FlickrLogos-32 [24] is a collection of images from Flickr containing 32 different logos. Most of the images contain a few and relatively clean, recognizable logo instances located near the center of an image compared to other datasets [11, 29]. The dataset is published to evaluate logo detection and recognition systems with 32 logo classes defined. The dataset has a total of 2,240 logo images, and it is partitioned into 10 training images, 30 validation images and 30 test images per class. It also contains 6,000 no-logo images to evaluate the false alarm rates of recognition systems. We cropped logo instances using bounding box annotations to evaluate our classification systems. In total, 3,372 logo instances were gathered by cropping.

BelgaLogos Dataset BelgaLogos [11, 18] is composed of 10,000 images from various aspects of everyday life with 37 logo classes annotated in a bounding box format. Unlike FlickrLogos-32, logos appear at diverse locations with large-scale variations, blur, saturation and occlusions. The quality levels of the samples are rated as either 'OK' or 'Junk' depending how clearly a sample is recognizable by human annotators. We cropped both 'OK' and 'Junk' logo instances to build a logo classification dataset. In total, 9,475 instances were collected. While FlickrLogos-32 shows an equal sample distribution per class, BelgaLogos shows a severe class imbalance from a small-sized class (2 samples) to a large-sized class (2,242 samples).

TopLogo-10 Dataset TopLogo-10 [28] contains 10 logo classes related to popular cloth, shoes and accessory brands. The images are collected from product images that are relatively clean and recognizable. Each class contains 70 images. We cropped logo instances using bounding box annotations and gathered a total of 853 logo samples. For the experiment, we defined a total of 11 logo classes by separating the 'Adidas' class into the 'Adidas-logo' and the 'Adidas-text' classes.

6.3. Embedding space

We provide t-SNE [20] plots using each method introduced in the main text. We select two representative evaluation scenarios, GTSRB→TT100K and Belga→Flickr32, for visualization. The result shows a clear difference between the feature distribution of VPE and the remaining feature spaces. It should be noted that VPE generates a more discriminative feature distribution compared to those by the competing approaches.

Siamese networks   Quadruplet networks
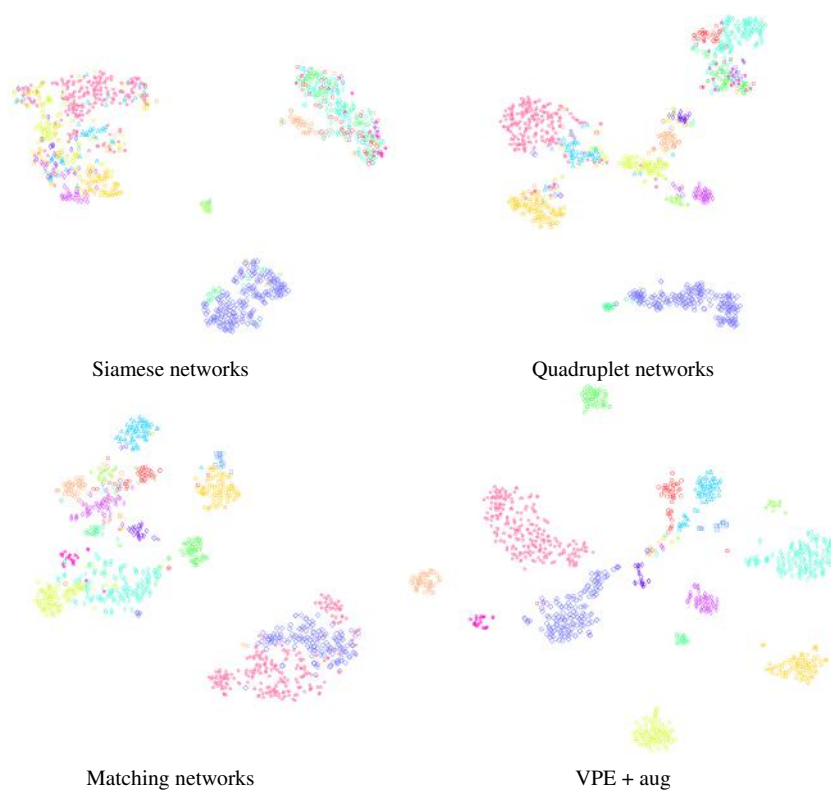
Matching networks   VPE + aug

Figure 8. t-SNE visualization of features on embedding space. Features are randomly sampled from 15 different unseen classes under the GTSRB→TT100K scenario for visualization.
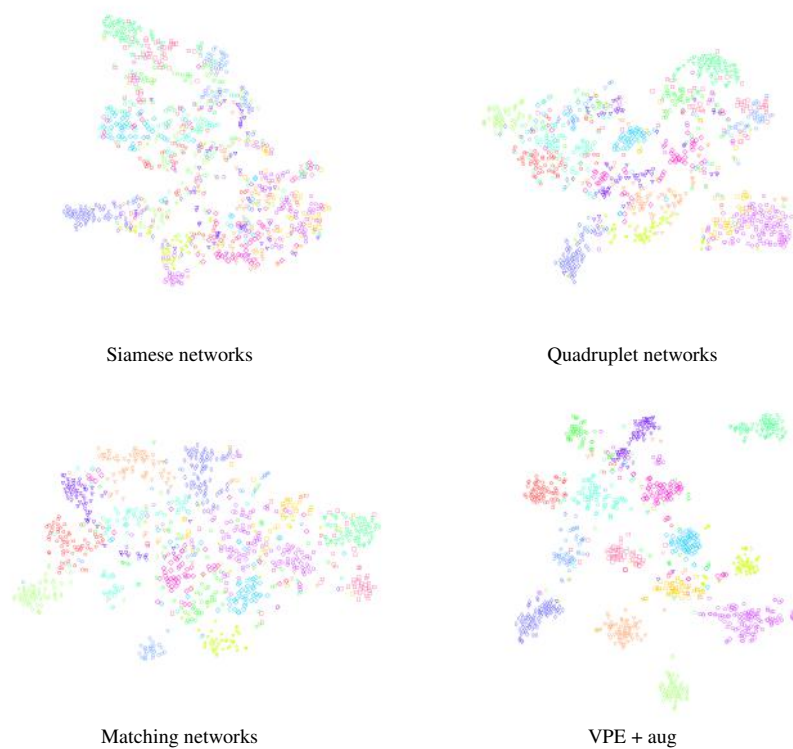

Siamese networks   Quadruplet networks

Matching networks   VPE + aug

Figure 9. t-SNE visualization of features on embedding space. Features are randomly sampled from 15 different unseen classes under the Belga→Flickr32 scenario for visualization.

6.4. Image retrieval test

We show more image retrieval results that could not be placed in the main text due to space constraints. The average images of the top 100 images retrieved by querying unseen prototypes in each scenario are displayed. The columns from left to right are the average images retrieved using the Siamese networks [15], Quadruplet networks [12], Matching networks [31] and by the proposed method.
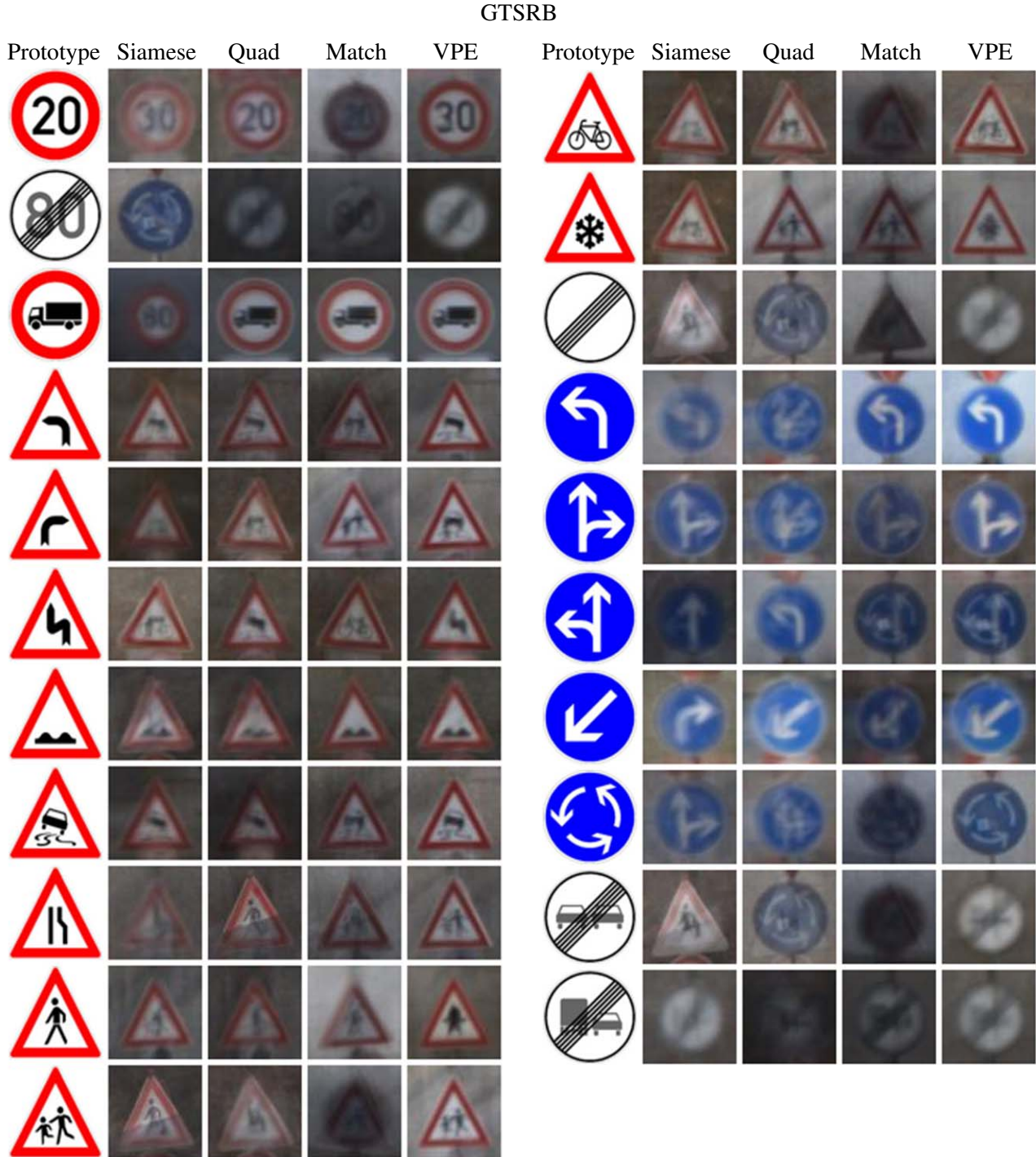

Figure 10. Average images of top 100 retrieved images by querying unseen prototypes in the GTSRB scenario.
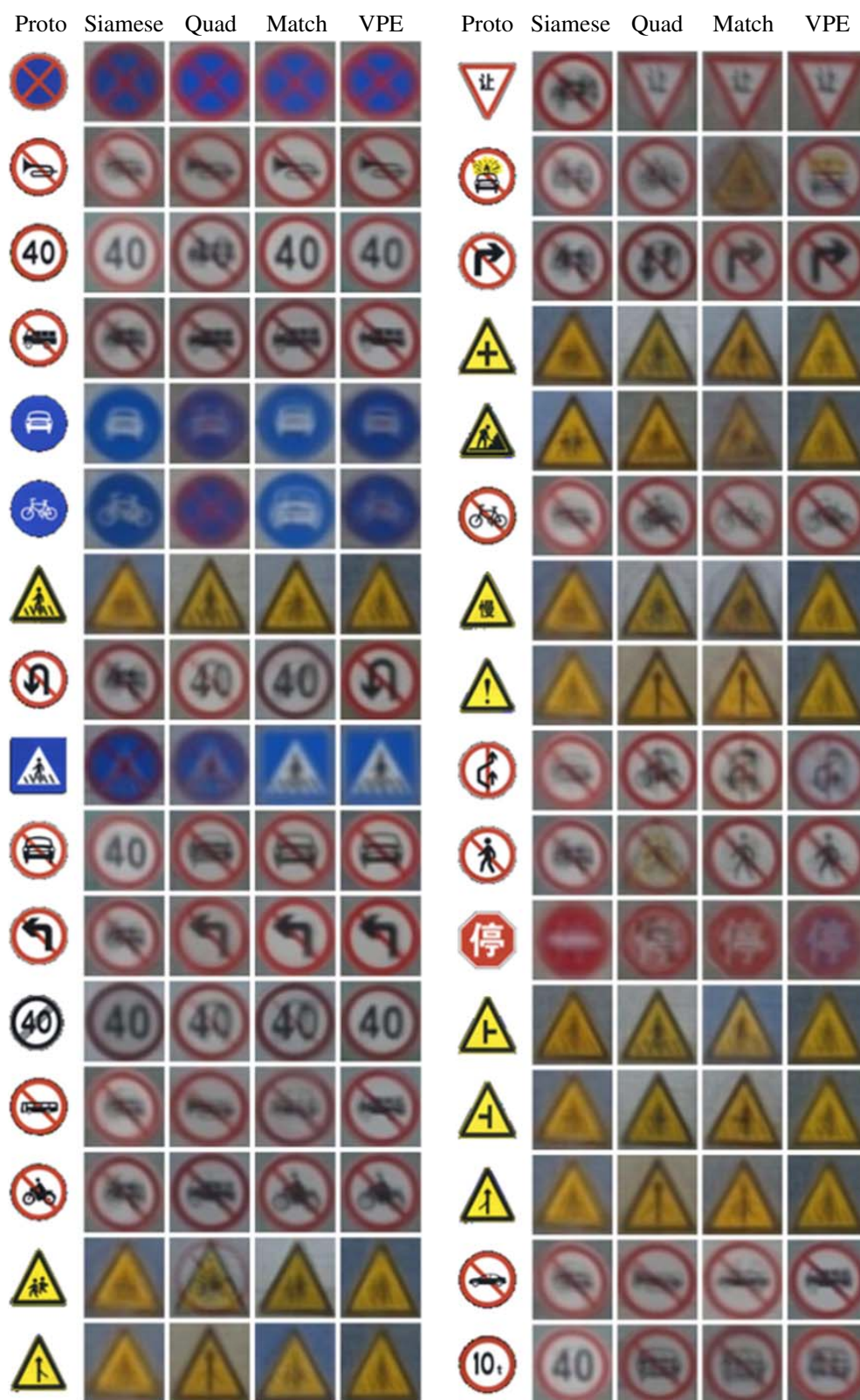
GTSRB→TT100K

| Proto | Siamese | Quad | Match | VPE | Proto | Siamese | Quad | Match | VPE |

Figure 11. Average images of top 100 retrieved images by querying unseen prototypes in the GTSRB→TT100K scenario.
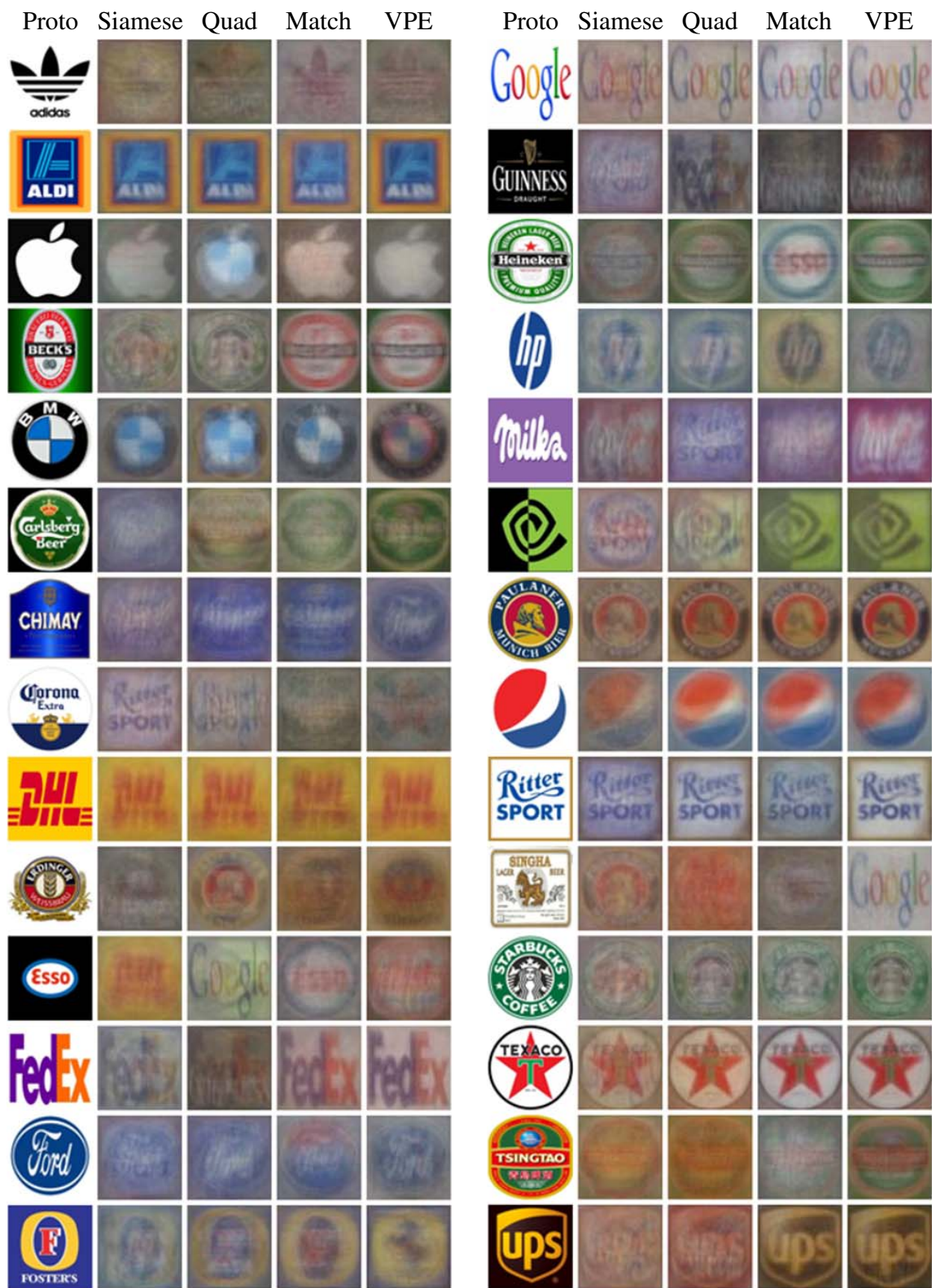
Belga→Flickr32

Figure 12. Average images of top 100 retrieved images by querying unseen prototypes in the Belga→Flickr32 scenario.