



CENTER FOR  
**Brains  
Minds+  
Machines**

CBMM Memo No. 62

March 13, 2017

# Discriminate-and-Rectify Encoders: Learning from Image Transformation Sets

**Andrea Tacchetti\*, Stephen Voinea\* and Georgios Evangelopoulos**

(\* denotes equal contribution)

Center for Brains, Minds, and Machines, MIT

McGovern Institute for Brain Research at MIT, Cambridge, MA, USA

## Abstract

The complexity of a learning task is increased by transformations in the input space that preserve class identity. Visual object recognition for example is affected by changes in viewpoint, scale, illumination or planar transformations. While drastically altering the visual appearance, these changes are orthogonal to recognition and should not be reflected in the representation or feature encoding used for learning. We introduce a framework for weakly supervised learning of image embeddings that are robust to transformations and selective to the class distribution, using sets of transforming examples (orbit sets), deep parametrizations and a novel orbit-based loss. The proposed loss combines a discriminative, contrastive part for orbits with a reconstruction error that learns to rectify orbit transformations. The learned embeddings are evaluated in distance metric-based tasks, such as one-shot classification under geometric transformations, as well as face verification and retrieval under more realistic visual variability. Our results suggest that orbit sets, suitably computed or observed, can be used for efficient, weakly-supervised learning of semantically relevant image embeddings.



This material is based upon work supported by the Center for Brains, Minds and Machines (CBMM), funded by NSF STC award CCF-1231216.

# Discriminate-and-Rectify Encoders: Learning from Image Transformation Sets

Andrea Tacchetti\*, Stephen Voinea\* and Georgios Evangelopoulos

The Center for Brains, Minds and Machines, MIT — McGovern Institute for Brain Research at MIT,

{atacchet, voinea, gevang}@mit.edu

\*denotes equal contribution

## Abstract

The complexity of a learning task is increased by transformations in the input space that preserve class identity. Visual object recognition for example is affected by changes in viewpoint, scale, illumination or planar transformations. While drastically altering the visual appearance, these changes are orthogonal to recognition and should not be reflected in the representation or feature encoding used for learning. We introduce a framework for weakly supervised learning of image embeddings that are robust to transformations and selective to the class distribution, using sets of transforming examples (orbit sets), deep parametrizations and a novel orbit-based loss. The proposed loss combines a discriminative, contrastive part for orbits with a reconstruction error that learns to rectify orbit transformations. The learned embeddings are evaluated in distance metric-based tasks, such as one-shot classification under geometric transformations, as well as face verification and retrieval under more realistic visual variability. Our results suggest that orbit sets, suitably computed or observed, can be used for efficient, weakly-supervised learning of semantically relevant image embeddings.

## 1. Introduction

The distribution of examples for a learning problem, such as visual object recognition, will exhibit variability across and within semantic categories. The former is due to the category-specific statistics; the latter is due to the variety of instances that share the same semantics and by transformations that preserve the identity, such as geometric or photometric changes. Such transformations will alter the properties of the visual scene but will not change the semantic category of an object. Recognition across novel views (position, size, pose), clutter and occlusions [8, 26, 19], and generalization to new examples from a category, are hallmarks of human and primate perception. Invariance to transformations has been consistently explored as the computational objective of representations for computer vision

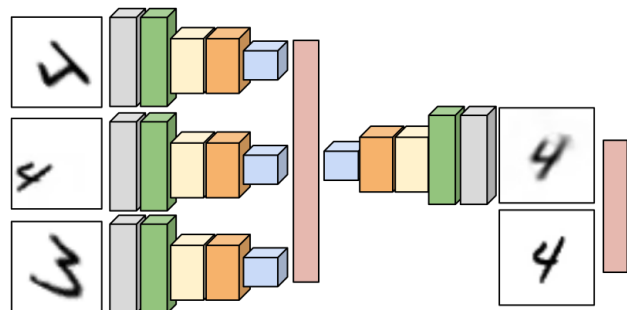


Figure 1: Discriminate-and-Rectify Encoder Network: Convolutional layers are followed by max pooling (cascade of two shown here). Orbit-Triplet (OT) loss is using only the encoding part for a triplet. Orbit-Encode (OE) reconstructs a canonical orbit element (lower right) from the deconvolutional decoder output. Tied weights are denoted by same colors, loss layers by red.

and machine learning [12, 13, 5, 20, 31, 7].

Representations that facilitate generalization in downstream supervised tasks can be learned in unsupervised or semi-supervised settings [1], where the distribution of observations is used for obtaining non-linear similarity metrics, reducing the dimensionality by disregarding nuisance directions or deriving interpretable, generative models. Unsupervised learning has been used, for example, for pre-training of neural networks with the goal of improving the convergence rates of end-to-end learning algorithms.

A question of theoretical and practical interest is under which conditions can representations learned without explicit supervision [4] match the performance of supervised learning methods that implicitly account for the representation of the data [18]. Learning from unlabeled (or implicitly labeled) data can be an alternative to using labeled examples for training multi-parameter deep neural networks, and building large-scale, generic and transferable learning systems economically. In addition, biological and cognitive learning paradigms, particularly in perceptual domains such as vision or speech, predict learning and generalization from a small number of labeled examples and an abundance of implicitly labeled observations or weak supervision.

A natural source of weak supervision is the formation of equivalence relations and classes in the input space, that are not necessarily related to the learning task. Such relations can be, for example, temporal, categorical or generative and partition the space in sets which we will loosely refer to as orbits in this paper. Example of orbits are the set of images of an object under rotations [1, 12] or the frames of a video of a moving object [25, 43, 24, 37]. This partition by orbit sets, in terms of granularity, lies in-between single-example and task-specific, semantic class partitions.

In this paper, we propose using orbit sets, defined by generic transformations, as weak supervision for learning representations in an invariant metric space. Two points are equivalent if one can be related to the other through a transformation, and the set of all equivalent points forms an orbit. Orbits are either explicitly generated (data augmentation) or implicitly specified (temporal continuity, data acquisition or association). As opposed to inference using an explicit transformation model [16] or factoring out the nuisance using explicit pooling [1], we learn deep, parametric embeddings using a novel loss function that incorporates the orbit equivalence relations.

The proposed *orbit metric loss* generalizes the triplet loss and denoising autoencoders to orbit sets, that promote approximate invariance and reconstruction. We separately study the two motivating special cases: the *orbit triplet loss*, a discriminative term which implicitly promotes invariance and selectivity with respect to examples drawn from the same or different orbits respectively, and the *orbit encoder loss*, a generative term, which learns to rectify or *de-transform* by mapping orbit points to a single, canonical element. The learned embeddings are compared, under the same parametrizations, to those from the supervised triplet loss [29], the surrogate class loss [10] and, when a full model of the orbit-generating process is available (e.g. affine transformations), spatial transformer networks [16].

The learned embeddings define robust metrics that are semantically relevant for distance-based and low supervised-sample regime tasks, such as ranking, retrieval, matching, clustering, graph-construction and relational learning. We provide quantitative comparisons on face verification and retrieval (on Multi-PIE dataset) and one-shot learning for classification (on MNIST with affine transformations). Our results show that partitioning the input space according to suitable orbit sets is a powerful weak supervision cue, which the proposed encoding loss can exploit effectively to learn semantically relevant embeddings.

## 2. Related work

Invariance to transformations that are orthogonal to the learning task has been the subject of extensive theoretical and empirical investigation in artificial and biological perception and recognition. A number of studies focused on

theoretical insights on the trade-off between invariance and selectivity through sufficient statistics [31], the optimality of explicit parametrizations with convolutions/pooling and memory-based learning [1, 2], as well as constructing invariants for compact groups and maps that are robust, rather than invariant, to diffeomorphisms [22]. These inspired feature extraction architectures for object recognition [9], texture classification [5], face verification [21], action recognition [33], and speech recognition [36, 41].

In this paper, we rely on the theoretical framework in [1, 2]. While relaxing some of the assumptions, e.g. compact groups, exact invariance, we make use of generic orbit sets, that can come from implicit supervision [1], or include non-group transformations, partial orbits and noisy samples. We propose a loss function that is a proxy for a margin-based invariance and selectivity in the representation, which can be used for end-to-end trainable encoders and does not rely on a particular parametrization, nor does it require access to a model of the orbit generation process.

Side information as a form of weak supervision has been employed for various distance metric learning algorithms [39]; for example, variants of the triplet loss function which rely on knowing which of two pairs corresponds to similar samples [38, 32]. Supervised versions of the triplet loss have been used for discriminatively-trained metric learning, through convolutional network parametrization, aiming to minimize the true objective of the task (e.g., face verification) [6, 29]. The triplet loss was also used for nonlinear dimensionality reduction and learning transformation-invariant embeddings [12]. Similar to the neighborhood graphs in [12], the orbits in our work can be obtained by side information or temporal proximity and are assumed known only for training.

Representation learning through surrogate classes populated by data augmentation transformations was explored in the exemplar CNN framework [10]. Autoencoder networks [14, 3, 4] have been typically used for dimensionality reduction (bottleneck features) and unsupervised pre-training of deep networks. Different reconstruction requirements or regularization terms have led to useful encodings by learning to perform denoising [35], sparse coding, contractive approximations for robustness [27], or respect temporal continuity (feature slowness) [43, 25]. Convolutional autoencoders enforce local spatial robustness through max pooling [42, 23]. Our method uses a combination of triplet loss and an explicit rectification accuracy loss through an encoder-decoder network.

The idea of using a representation of the transformations for robust embeddings has been explored through explicit estimation of the parameters of an exact model of the generative process in spatial transformer networks [16], estimation of a latent representation in transforming autoencoders [13], or a distributed representation [42]. Similar to [28],

the proposed orbit loss functions can be used as a regularizer of the discriminative loss of a deep network for representations that are both robust and discriminatively trained.

### 3. Background

We begin by reviewing relevant background concepts in order to provide context and formulate clearly the proposed losses and weakly-supervised learning methods.

**Learning and representations** The feature map or data representation  $\Phi: \mathcal{X} \rightarrow \mathcal{F}$  for a learning problem on input space  $\mathcal{X}$ , can be explicitly selected or learned using principles such as

- distance preservation  $\|\Phi(x) - \Phi(x')\|_{\mathcal{F}} \approx \|x - x'\|$  or contraction  $\|\Phi(x) - \Phi(x')\|_{\mathcal{F}} \leq l \|x - x'\|, l \in [0, 1)$ ,
- reconstruction  $\|x - \tilde{\Phi} \circ \Phi(x)\| \leq \epsilon, \tilde{\Phi}: \mathcal{F} \rightarrow \mathcal{X}$ ,
- invariance and selectivity  $x' \sim x \Leftrightarrow \Phi(x) = \Phi(x')$ ,

where  $\sim$  denotes equivalence and  $\circ$  function composition, i.e.  $(\tilde{\Phi} \circ \Phi)(x) = \tilde{\Phi}(\Phi(x))$ .

**Remark 1.** *The feature map  $\Phi$  is selected from some hypothesis space of functions  $\mathcal{H} \subseteq \{\Phi \mid \Phi: \mathcal{X} \rightarrow \mathcal{F}\}$ . In practice, some parametrization of the elements of  $\mathcal{H}$  that renders the resulting representation learning problem tractable is necessary.*

**Remark 2.** *For kernel machines (or shallow networks),  $\Phi$  is preselected, or implicitly induced by a kernel function  $K: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ ,  $K(x, x') = \langle \Phi(x), \Phi(x') \rangle, \forall x, x' \in \mathcal{X}$ . For deep networks,  $\Phi$  is parametrized, typically through linear projections and non-linearities, and jointly learned with the predictor function. Moreover, it involves multiple maps  $\Phi_l: \mathcal{F}_{l-1} \rightarrow \mathcal{F}_l$  in the form of compositions of multiple representation layers  $\Phi = \tilde{\Phi}_L \circ \dots \circ \tilde{\Phi}_1$ .*

**Metric learning** In the general case, the global metric learning problem [17] is learning a distance function  $D(x, x')$  between two points  $x, x' \in \mathcal{X}$  as the distance  $D_{\mathcal{F}}(\Phi(x), \Phi(x'))$  in a new space  $\mathcal{F}$ :

$$D(x, x') = (\Phi(x) - \Phi(x'))^T (\Phi(x) - \Phi(x')). \quad (1)$$

The representation  $\Phi: \mathcal{X} \rightarrow \mathcal{F}$  can express linear, kernelized or nonlinear mappings and is obtained by the solution of a regularized, constrained minimization problem, using some form of side information, such as the similarity between pairs [39] or triplets of points [32].

**Triplet loss** The large margin nearest neighbor loss [38] was developed for supervised learning of a distance metric by pulling together and pushing apart same- and different-class neighbors, respectively. The closely related contrastive

loss [6] uses pairs of observations  $(x_i, x_j)$  and their label  $(y_i, y_j)$  agreement to decrease or increase their distance  $D(x_i, x_j) = \|\Phi(x_i) - \Phi(x_j)\|_{\mathcal{F}}^2$  by learning  $\Phi$  through:

$$\min_{\Phi \in \mathcal{H}} \sum_{\substack{i, j \neq i \\ i, j \in \mathcal{H}}}^n \mathbb{1}_{y_i=y_j} D(x_i, x_j) + (1 - \mathbb{1}_{y_i=y_j}) |\alpha - D(x_i, x_j)|_+, \quad (2)$$

where  $n$  is the size of the training set  $\{(x_i, y_i)\}_1^n$ ,  $\alpha \in \mathbb{R}_+$  a distance margin for the non-matching pairs and  $|\alpha|_+ = \max\{0, \alpha\}$  the hinge loss function. The triplet loss is based on defining point triplets using their label agreement  $\mathcal{T} = \{(x_i, x_p, x_q) \mid y_i = y_p, y_i \neq y_q\}$  [29] and aims to enforce

$$D(x_i, x_p) \leq D(x_i, x_q) - \alpha \quad (3)$$

by minimizing the mismatch part of the large margin loss:

$$\min_{\Phi \in \mathcal{H}} \sum_{i=1}^{|\mathcal{T}|} |D(x_i, x_p) + \alpha - D(x_i, x_q)|_+. \quad (4)$$

**Autoencoders** An autoencoder is composed of an encoding map  $\Phi: \mathbb{R}^d \rightarrow \mathbb{R}^k$  and a decoding map  $\tilde{\Phi}: \mathbb{R}^k \rightarrow \mathbb{R}^d$ , where we assume  $\mathcal{X}$  and  $\mathcal{F}$  be Euclidean spaces and  $\mathcal{H}$  and  $\tilde{\mathcal{H}}$  to be the appropriate hypothesis spaces, learned by minimizing a reconstruction loss

$$\min_{\Phi \in \mathcal{H}, \tilde{\Phi} \in \tilde{\mathcal{H}}} \sum_{i=1}^n L(x_i, \tilde{\Phi} \circ \Phi(x_i)) \quad (5)$$

where  $L: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}_+$  is typically the square or cross-entropy loss. The encoding is parametrized by a linear map using  $k$  projection units or filters  $W = (w_1, \dots, w_k) \in \mathbb{R}^{d \times k}$ , an offset  $b \in \mathbb{R}^d$ , and a nonlinear function  $\sigma: \mathbb{R} \rightarrow \mathbb{R}$  applied element-wise

$$\Phi(x) = \sigma \circ (Wx + b). \quad (6)$$

The decoding map is typically of a similar form

$$\tilde{\Phi} \circ \Phi(x) = \sigma \circ (\tilde{W}\Phi(x) + \tilde{b}) \quad (7)$$

usually constrained having tied-weights,  $\tilde{W} = W^T$ , for a reduced number of parameters. Both maps can have multiple layers and learned with additional priors through regularization on  $W$  or the activations of the hidden layers [4, 27], reconstructing perturbations of  $x$  [35] or convolutional structure on  $W$  [40, 23, 42] and pooling.

**Transformations and orbit sets** Consider a family of transformations as a set of maps  $\mathcal{G} \subset \{g \mid g: \mathcal{X} \rightarrow \mathcal{X}\}$ . We will denote by  $gx$  the action of the transformation represented by  $g$  on  $x$ , which generates point  $x'$ , i.e.  $x' = gx = g(x)$ . The transformations can be parametrized by  $\theta_j \in \Theta$ , such that  $\mathcal{G} = \{g_j = g_{\theta_j} \mid \theta_j \in \Theta\}$ . The set can have algebraic structure, e.g. form a group [2, 22, 7] (Fig. 2, row 1).



Figure 2: Examples of transformation orbits obtained through image rotations/data augmentation (row 1: MNIST, digits) and continuity (row 2: MulitPIE, faces).

**Definition 1** (Group orbits [2]). *An orbit associated to an element  $x \in \mathcal{X}$  is the set of points that can be reached under the transformations  $\mathcal{G}$ , i.e.,  $\mathcal{O}_x = \{gx \in \mathcal{X} | g \in \mathcal{G}\} \subset \mathcal{X}$ . Given a group structure on  $\mathcal{G}$ , the transformations partition the input space  $\mathcal{X}$  into orbits by defining equivalence relations:  $x \sim x' \Leftrightarrow \exists g \in \mathcal{G} : x' = gx, \forall x, x' \in \mathcal{X}$ . As a result, each  $x \in \mathcal{X}$  belongs to one and only one orbit  $\mathcal{O}_x = \mathcal{O}_{gx}, \forall g \in \mathcal{G}$  and the input space is  $\mathcal{X} = \cup_{x \in \mathcal{X}, g \in \mathcal{G}} \mathcal{O}_{gx} = \cup_{x \in \mathcal{X}} \mathcal{O}_x$ . Using the fact that orbit are sets defined by equivalence relations in  $\mathcal{X}$ , we can extend the definition to relations or set memberships provided by categorical labels.*

**Definition 2** (Generic orbits). *An orbit associated to an element  $x \in \mathcal{X}$  is the subset of  $\mathcal{X}$  that includes  $x$  along with an equivalence relation, i.e. the equivalence class  $\mathcal{O}_x = \{x' \in \mathcal{X} | x \sim x'\} \subset \mathcal{X}$ . The equivalence relation is given by a function  $c : \mathcal{X} \rightarrow \mathcal{C}$  such that  $x \sim x' \Leftrightarrow c(x) = c(x')$ . Examples of such maps are the labels of a supervised learning task, the indexes of vector quantization codewords or, for the case of sequential data such as videos, the sequence membership, with  $\mathcal{C}$  the set of classes, codewords or sequences respectively (Fig. 2, row 2).*

**Surrogate classes and exemplar loss** The Exemplar loss, introduced as a way to combine data augmentation and weak supervision for training convolutional networks [10], uses a surrogate class for each point in an unlabeled training set  $\mathcal{X}_n = \{x_i\}_{i=1}^n$ . The surrogate class instances are generated by random transformations, sampled from  $\mathcal{G}$ , of the class prototypes  $\{g_j x_i\}_{j=1}^{k_i \leq |\mathcal{G}|}$ . An embedding  $\Phi$  is learned by minimizing a discriminative loss with respect to the surrogate classes:

$$\min_{\Phi \in \mathcal{H}, f \in \mathcal{H}_f} \sum_{i=1}^n \sum_{j=1}^{k_i \leq |\mathcal{G}|} L(i, f(\Phi(g_j x_i))) \quad (8)$$

where  $i$  indexes the original, untransformed training set  $\mathcal{X}_n$  and serves as the surrogate class label for all points generated from  $x_i$ ;  $f$  is a classifier learnt jointly with the embedding.

**Spatial transformer networks (STNs)** When a plausible forward model of the process that generates orbits is known, i.e. when a suitable parametrization of  $\mathcal{G} = \{g_j = g_{\theta_j} | \theta_j \in \Theta\}$  is available, STNs [16] are trainable modules that learn to undo a transformation in  $\mathcal{G}$ , by explicitly

transforming the input of a feature map. STNs introduce a specific modification to the parametrization of  $\Phi$ , that for an input  $g_{\theta} x_i$ , provides an estimate of  $\theta$  and applies the inverse transformation  $g_{\bar{\theta}} \circ g_{\theta} x_i \approx x_i$ . This module acts as an *oracle* that provides a rectified, untransformed version of its input, which is then passed to downstream embedding maps. The resulting embedding is robust, by construction, to transformations in  $\mathcal{G}$ .

## 4. Metric learning with orbit loss

We introduce a novel loss function for learning an embedding for a distance metric using as weak supervision the set memberships on transformation orbits. The loss aims to jointly, adaptively and in a data-driven manner enforce *invariance*, to the transformations captured by the orbit sets, *selectivity* and *low rectification error* on the representation. The loss function is independent of the embedding parametrization, though it implies a siamese (tied weights) and an encoder-decoder network architecture (Fig. 1). In this paper we will learn deep, convolutional encodings. Orbit sets are obtained either from explicit transformations of the unlabeled input data, e.g. each sample generates an orbit, or from a weak supervision signal involving data continuity, e.g. subsets of the training set correspond to data collected sequentially or under multiple views.

### 4.1. Problem statement

Let training set  $\mathcal{X}_n = \{x_i\}_{i=1}^n \in \mathcal{X}$  be a set of unlabeled instances. We assume the input to be in  $\mathcal{X} = \mathbb{R}^d$ , for example having  $x_i$  be the vectorized intensity values of an image. We aim to learn a feature map or embedding  $\Phi : \mathcal{X} \rightarrow \mathcal{F}$ , in space  $\mathcal{F} = \mathbb{R}^k$  with  $k \leq d$ , such that the metric  $D : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}_+$  given by the distance

$$D(x, x') = \|\Phi(x) - \Phi(x')\|_{\mathcal{F}}^2, \quad (9)$$

where  $\|\cdot\|_{\mathcal{F}}$  denotes the norm in  $\mathcal{F}$ , is invariant and selective with respect to the transformations of  $x$  captured by an orbit set  $\mathcal{O}_x$ , equivalently written as:

$$x' \sim x \Leftrightarrow x, x' \in \mathcal{O}_x \Leftrightarrow D(x, x') \leq \epsilon, \quad (10)$$

where we relax the exact invariance condition using an  $\epsilon$ -approximation to the zero-norm distance. In terms of  $\Phi$ , Eq. (10) defines a sufficient and necessary condition for two points being equivalent under the transformation in the orbit set [2]. Note that the requirement for selectivity, i.e., the converse direction, makes  $(\mathbb{R}^d, D)$  a proper metric space with an invariant, in this  $\epsilon$ -approximation sense, metric.

### 4.2. Orbit sets

The definition of the orbit sets  $\mathcal{O}_x$  is crucial for the proposed framework and can be based on the data distribution

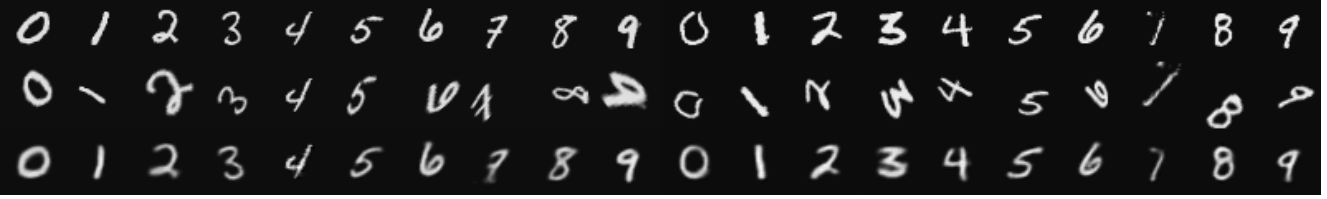


Figure 3: Rectification examples from the orbit decoder output for samples from affine MNIST, showing canonical image  $x_c$  (top), randomly transformed  $x_i = g_i x_c$  (middle) and rectified output  $\tilde{\Phi} \circ \Phi(x_i)$  (bottom).

and the learning problem; here we give a few examples of orbit sets.

**Augmentation** Given a parametrized family of transformations  $\mathcal{G}$ , one can generate orbit samples for a given  $x$  by randomly sampling from the parameter vectors  $\{\theta_j \in \Theta\}$  and letting  $\mathcal{O}_x = \{g_{\theta_j} x \mid \theta_j \in \Theta\}$ . Examples include geometric transformations (rotation, translation, scaling), e.g. Fig 2 (row 1), or typical data augmentation transforms (cropping, contrast, color, blur, illumination etc.) [10].

**Acquisition** If the data acquisition process is part of the learning problem, e.g. online/unsupervised learning, or included as meta-data, e.g. multiple samples of an object across time, conditions or views, e.g. Fig 2 (row 2), then an orbit can be associated to all samples from the same sequence or session [11].

**Temporal continuity** For sequential data such as videos, an orbit can be a continuous segment of the video stream, following plausible assumptions on feature smoothness and continuity of the representation in time [37, 43].

### 4.3. Orbit metric loss

Assume the set of orbits  $\{\mathcal{O}_{x_i}\}$  given, either via an a priori partition of the training set  $\mathcal{X}_n$  in a number of equivalence classes such that  $\mathcal{X}_n = \cup_{x_i} \mathcal{O}_{x_i}$ , or by augmentation of each  $x_i \in \mathcal{X}_n$  such that  $\mathcal{O}_{x_i} = \{g_{\theta_j} x \mid \theta_j \in \Theta\}$ . Given the orbits, consider a set of triplets

$$\mathcal{T} \subset \{(x_i, x_p, x_q) \mid x_i \in \mathcal{X}_n, x_p \in \mathcal{O}_{x_i}, x_q \in \mathcal{O}_{x_q} \neq \mathcal{O}_{x_i}\} \quad (11)$$

such that each  $x_i$  is assigned a *positive example*  $x_p$  (in-orbit), i.e.  $x_i \sim x_p \Leftrightarrow x_i, x_p \in \mathcal{O}_{x_i}$  and a *negative example*  $x_q$  (out-of-orbit), i.e.  $\mathcal{O}_{x_q} \neq \mathcal{O}_{x_i}$ . We further assume that each orbit is equipped with a *canonical example*  $x_c \in \mathcal{O}_{x_i}$ .

**Definition 3** (Orbit canonical element). *An orbit point that provides a reference coordinate system for the family of transformations that generate the orbit. For  $\mathcal{O}_x$  obtained through a generative process applied on  $x$ ,  $x_c$  is the output of the identity transformation  $g_0 \in \mathcal{G}$ , i.e.  $x_c = g_0 x$ . For orbits from categorical meta-data,  $x_c$  is empirically chosen to be the ‘regular’ view or neutral condition (Fig. 3, row 1).*

The proposed loss function, reflected in the architecture in Fig. 1, is composed of two terms; a discriminative term  $L_t(x_i, x_p, x_q)$ , based on the triplet loss, using distances be-

tween the encodings  $\Phi : \mathbb{R}^d \rightarrow \mathbb{R}^k$  on the feature space  $\mathbb{R}^k$ ; a reconstruction error  $L_e(x_i, x_c)$  between a decoder  $\tilde{\Phi} : \mathbb{R}^k \rightarrow \mathbb{R}^d$  output and the canonical, as a distance on the input space  $\mathbb{R}^d$ :

$$\min_{\Phi, \tilde{\Phi}} \sum_{x_i \in \mathcal{X}_n} \frac{\lambda_1}{d} L_t(t_i) + \frac{\lambda_2}{k} L_e(x_i, x_c), \quad t_i = (x_i, x_p, x_q)$$

$$L_t(t_i) = \left\| \|\Phi(x_i) - \Phi(x_p)\|_{\mathbb{R}^k}^2 + \alpha - \|\Phi(x_i) - \Phi(x_q)\|_{\mathbb{R}^k}^2 \right\|_+$$

$$L_e(x_i, x_c) = \left\| x_c - \tilde{\Phi} \circ \Phi(x_i) \right\|_{\mathbb{R}^d}^2. \quad (12)$$

The constants  $\lambda_1$ ,  $\lambda_2$  and  $\alpha$  control the relative contribution of each term and the distance margin. The loss is independent of the parameterization of  $\Phi$ ,  $\tilde{\Phi}$  but depends on the selection of the triplets  $\mathcal{T} = \{t_i\}$  for  $\mathcal{X}_n$ , given the orbits, and the canonical instance  $x_c$  for each orbit.

### 4.4. Orbit triplet loss

For  $\lambda_2 = 0$  the orbit metric loss reduces to the triplet loss in Eq. (4), when similarity and dissimilarity are specified by orbit memberships. Points that lie on the same orbit are pulled together and points on different orbits are pushed apart. The minimizer will be pushed to satisfy Eq. (3), using all triplets in the training set. Note that in the theoretical minimum of  $L_t$ , e.g. using the subgradient of the hinge loss, Eq. (3) is satisfied. The orbit triplet loss follows a *Siamese network* architecture [6], with a tied-weight embedding trained using triplets as input.

The following proposition shows how, for the case of bounded-norm embeddings, minimizing the triplet loss, thus pushing to minimize Eq. (3), leads to an operational definition of selective robustness to transformations (invariance) with a tolerance margin  $\alpha$ .

**Proposition 1.** *Let  $\Phi$  be in the space of functions with norm bounded by  $\sqrt{2\alpha}$ , i.e.  $\Phi \in \mathcal{H} \subset \{\Phi \mid \Phi : \mathcal{X} \rightarrow \mathcal{F}, \|\Phi\|_{\mathcal{F}}^2 \leq 2\alpha\}$ . If Eq. (3) is true, then  $\Phi$  is invariant for the orbit transformations and selective for the orbit identities, according to the  $\epsilon$ -approximate definition in Eq. (10) with  $\epsilon = \alpha$ .*

*Proof.* For a triplet  $(x, x_p, x_q)$ , with  $x_p, x_q$  being positive and negative examples for  $x$ , Eq. (3) gives  $D(x, x_q) \geq D(x, x_p) + \alpha \geq \alpha, \forall x, x_p \in \mathcal{X}$  and  $D(x, x_p) \leq D(x, x_q) -$

$\alpha \leq \alpha$ , as by the bounded norm assumption  $D(x, x_q) = \|\Phi(x) - \Phi(x_q)\|_{\mathbb{R}^k}^2 \leq 2\alpha, \forall x, x_q \in \mathcal{X}$ . Since  $x_p, x_q$  are same and different orbit elements, it holds that  $x \sim x_p, x \approx x_q \Leftrightarrow D(x, x_p) \leq \alpha$ , i.e.  $\Phi$  satisfies (10) with  $\epsilon = \alpha$ .  $\square$

#### 4.5. Orbit encoder

For  $\lambda_1 = 0$  the orbit metric loss reduces to a loss that penalizes, using an additional decoder map  $\tilde{\Phi} : \mathcal{F} \rightarrow \mathcal{X}$ , the reconstruction error between the output of point  $x_i$  to the canonical  $x_c$  of the orbit  $\mathcal{O}_{x_i}$ . This is also the error of the transformation rectification that  $\tilde{\Phi} \circ \Phi : \mathcal{X} \rightarrow \mathcal{X}$  applies on the input  $x_i$ , assumed to be the transformation of  $x_c$ . This loss is a novel type of autoencoder loss that learns to de-transform the input.

The motivation is the generalization of denoising autoencoders, that learn to reconstruct clean versions of their noisy input, to transformations with or without an explicit generative model, using the equivalence of points within an orbit. *Orbit encoders* learn to de-transform an input adaptively, for all transformations in the training set orbits, by mapping points onto a pre-selected canonical orbit element (Fig. 3, top row). This provides a reference point for the set, such that every point in  $\mathcal{O}_{x_i}$  can be seen as  $x_i = c_i(x_c)$  or  $x_i = g_i x_c$  for a known transformation process. The reconstruction error is then  $\|x_c - \tilde{\Phi} \circ \Phi(g_i x_c)\|$  and the minimization pushes the solution towards an ‘inversion’ of the transformation  $\tilde{\Phi} \circ \Phi \approx g_i^{-1}$ , jointly for all points in the training set. Another way to see the rectification objective is as trying to reconstruct any given  $x_c$  from an artificially transformed version of it  $x_i = g_i x_c$ .

Training requires pairs  $(x_c, x_i)$  and the choice of the canonical for each orbit has to be consistent only across the same semantic class of a downstream task, e.g. all orbits of the same class. The loss enforces selectivity on  $\Phi$  by preserving sufficient information to reconstruct the input irrespective of the transformation.

#### 4.6. Parameterization of the embedding

The mapping  $\Phi$  is parameterized through multiple layers

$$\Phi = \Phi_L \circ \dots \circ \Phi_1, \Phi_l : \mathcal{F}^{l-1} \rightarrow \mathcal{F}^{l+1}, \mathcal{F}^0 = \mathbb{R}^d, \mathcal{F}^L = \mathbb{R}^k \quad (13)$$

with each one being a  $k_l$ -dimensional feature map of linear projections on  $f_l$  filters and nonlinearities of the form in Eq. (6). The output of layer  $l$  given layer  $l-1$  is

$$\Phi_l \circ \Phi_{l-1}(\cdot) = \sigma \circ (W_l \Phi_{l-1}(\cdot) + b_l), \quad (14)$$

where  $W_l \in \mathbb{R}^{k_l \times k_{l-1}}$  the  $k_l \times k_{l-1}$  weight matrix and  $b_l \in \mathbb{R}^{k_l}$ . We consider convolutional maps, where groups of filters correspond to the same local kernel shifted over the support of the input, i.e. each filter is sparse on the input (local connectivity) and the projection is a convolution

operator (weight sharing). The activation in Eq. (14) is then

$$\Phi_l \circ \Phi_{l-1}(\cdot) = \sigma \circ (W_l \star \Phi_{l-1}(\cdot) + b_l) \quad (15)$$

where  $\star$  denotes convolution with each row of  $W_l \in \mathbb{R}^{f_l \times k_{l-1}}$  (with  $k_l = f_l \times c_l$ , where  $c_l$  the number of shifts of the convolution kernel) and  $b_l \in \mathbb{R}^{f_l}$  (one bias per channel).

For the nonlinearity  $\sigma$  we use the hard rectifier (ReLU activation functions) given by  $\sigma(a) = |a|_+ = \max\{0, a\}, a \in \mathbb{R}$ , which is applied element wise on the pre-activation output  $h_l = (W_l \star \Phi_{l-1}(\cdot) + b_l) \in \mathbb{R}^{k_l}$ , i.e.  $(\sigma \circ h_l)^j = \sigma(h_l^j), j = 1 \dots, k_l$ . Batch normalization is applied on  $h_l$  before  $\sigma$  as  $(\eta \circ h_l)^j = \eta(h_l^j) = \gamma^j h_l^j + \beta^j$ , where  $\gamma$  and  $\beta$  are trainable parameter vectors and each dimension of  $h_l$  is standardized to be zero mean and unit variance, using the statistics of the training mini-batch [15].

In addition, max pooling nonlinearities are introduced after a number of convolution layers in order to increase spatial invariance and decrease the feature map sizes. For each filter  $j$ , the layer is looking at the corresponding support in its input  $\Phi_{l-1} \mathbb{1}_{f_l=j}$  and takes the maximum over sets of convolution values defined on a grid  $\mathcal{N}$  of neighboring values, i.e.  $\max\{\Phi_{l-1} \mathbb{1}_{\mathcal{N}, f_l=j}\}$ .

The decoder  $\tilde{\Phi}$  is a series of deconvolution and unpooling layers [40], in direct correspondence to the encoder in number of layers, units per layer, filters, size of kernels, and with tied weights such that  $\tilde{W}_l = W_l^T, l = 1, \dots, L$ .

## 5. Experiments

We compared the embeddings learned using the proposed loss, Orbit Joint (OJ) in Eq. (12) and the two special cases, Orbit Triplet (OT) ( $\lambda_2 = 0$ ) and Orbit Encode (OE) ( $\lambda_1 = 0$ ), and three reference, closely related methods: Supervised Triplet (ST) [29], Exemplar (EX) [10], and standard Autoencoder (AE). Each loss was used for learning a map from the input to a metric space using the same network/parametrization and varying degrees of supervision (unsupervised, supervised or weakly-supervised using the set of orbits). Once the embeddings were learned, the *training and test sets* for the downstream tasks (one-shot digit classification on affine MNIST, face verification and retrieval on Multi-PIE) were encoded and used to evaluate the performance. The *embedding set* used for training the networks, i.e. the collection of orbits, was kept separate from any data used in the downstream tasks. For the affine-MNIST evaluations, we also compared our methods to an embedding parametrization featuring a Spatial Transformer Networks module [16], trained with orbit supervision (OT-STN) or full supervision (ST-STN).

### 5.1. Network and training details

The encoder was a deep convolutional network following the VGG architecture [30]. Each layer was a series of

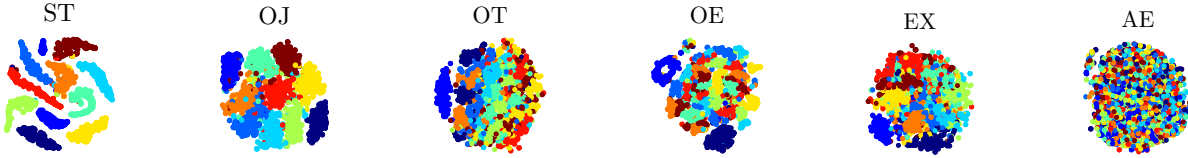


Figure 4: 2D t-SNE visualizations of a subset of the affine MNIST test set (original set augmented by random affine transformations), under different embeddings. Colors code the 10 different semantic classes (digit identity).

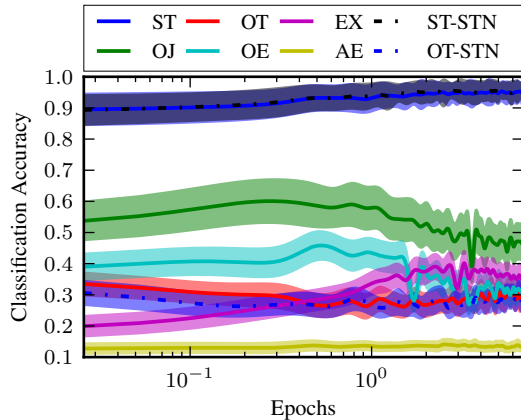


Figure 5: Nearest Neighbor classification using one example-per-class on MNIST augmented with affine transformations. Accuracies are averaged over 100 random re-samples of the training set (shaded region 1 s.d.) and shown at different times during the training of the embeddings.

convolutions with a small  $3 \times 3$  kernel (of stride 1, padding 1), batch normalization [15] and ReLU activation. A spatial max pooling layer (of stride 2 and size either  $2 \times 2$  or  $4 \times 4$ ) was used every two such layers of convolutions. The number of channels doubled after each max pooling layer, ranging from 16 to 128 for MNIST and 64 to 512 for Multi-PIE. Four iterations of convolution and pooling were followed by a final fully-connected layer of size 1024. The decoder was a deconvolutional network, reversing the series of operations in the encoder using convolutional reconstruction and max unpooling [40]. Encoder and decoder weights were tied with free biases. Training was done with minibatch Stochastic Gradient Descent using the ADAM optimizer. For MNIST experiments we used minibatches of 256, and for Multi-PIE, 72. The selection of the triplets followed the soft negative selection process from [29]. The values for  $\lambda_1$  and  $\lambda_2$  were set equal in these experiments, but they can be selected by cross-validation. The STNs modules, used for the affine-MNIST comparisons, consisted of two max pooling-convolution-ReLU blocks with 20 filters of size  $5 \times 5$  (stride 1), pooling regions of size  $2 \times 2$  and no overlap, and followed by two linear layers.

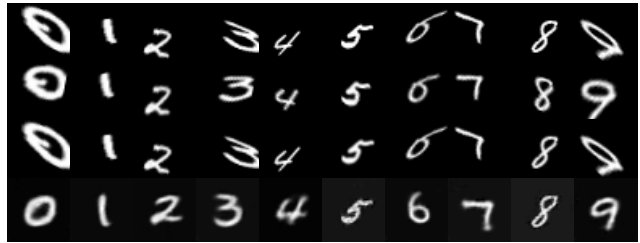


Figure 6: Input (row 1) and rectifications for ST-STN (row 2), OT-STN (row 3) and OJ (row 4). STNs produced convincing rectifications when full supervision was available (ST-STN) but learned the identity with weak supervision (OT-STN). The encoder-decoder map trained with the proposed orbit metric loss produces accurate rectifications using only weak supervision.

## 5.2. Affine transformations: MNIST

We created a version of MNIST, using 32 random affine transformation for each point in the original MNIST dataset (samples in Fig. 3, middle row). Transformations were sampled uniformly from the union of the following intervals: rotation from  $[-90^\circ, 90^\circ]$ , shearing factor from  $[-0.3, 0.3]$ , scale factor from  $[0.7, 1.3]$ , and translation in each dimension from  $[-15, 15]$  pixels. The orbit set consisted of the original MNIST training set ( $50 \times 10^3$  images), augmented by 32 transformations for each sample, resulting in a total of  $1650 \times 10^3$  images, grouped in  $50 \times 10^3$  orbits. Each orbit, of size 33, is the set of a single original image (canonical) and the corresponding random transformations of it.

The learned embeddings using this set were employed in a one-shot classification task to assess their invariance and selectivity properties. The training set consisted of 10 images, one from each semantic class. These were drawn at random from the original MNIST validation (augmented by 32 random affine transformations). The test set consisted of  $25 \times 10^3$  images randomly drawn from the original MNIST test set (plus transformations). Figure 4 shows the 2D t-SNE plots [34] of the learned embeddings on a random subset of the test set. Qualitatively, the best separation and grouping was observed with the fully supervised triplet loss (ST), followed by the weakly supervised orbit joint loss (OJ).

Nearest Neighbor classification was used for predicting the label of each test point from the 10 image labelled training set, which is not controlled for transformations. Fig-



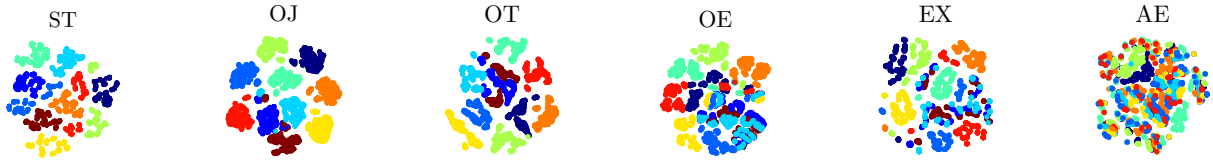


Figure 7: 2D t-SNE visualizations of the embeddings for face images from 10 subjects/classes (coded by different colors), randomly sampled from Multi-PIE test set.



Figure 8: Examples of retrieval and rectification on Multi-PIE. Rows correspond to: 1) canonical pose of the seed image, 2) seed image, 3) Top-1 retrieved using the Orbit Joint (OJ) embedding, 4) Top-1 retrieved using the Exemplar (EX) embedding [10] and 5) the reconstructed canonical pose using the decoder of the OJ network.

Figure 5 shows classification accuracy results during embedding training epochs. At each iteration, the accuracy is shown as mean with standard deviation (sdtv) error bars over 100 different labelled set selections. As expected, the supervised ST performed best and the unsupervised AE was the lower baseline. Of the weakly-supervised methods, the orbit metric loss OJ achieved the top accuracy, followed by OE and EX. Spatial transformer network modules provided a small improvement in accuracy (2%, consistent with the improvement reported in [16]) when used with full supervision (ST vs. ST-STN). However, when only orbit information was available (OT vs. OT-STN), there was no difference in performance. This is further reflected in Fig. 6 which shows rectification examples from the output of the STN module and the learned decoder with our method.

### 5.3. Face transformations: Multi-PIE

The Multi-PIE dataset [11] contains images of faces of 129 individuals, captured from 13 distinct viewpoints and under 20 different illumination conditions. Acquisition was carried out across four sessions, resulting in a dataset of  $129 \times 13 \times 20 \times 4 = 134160$  images. For learning the maps from the input to the metric space, we used all images from three of the sessions to form the embedding sets and left out all images from the fourth session for performance assessment in the downstream task. During training of the embedding map  $\Phi$ , the ST method had access to the face identity for each image, thus considering all images of the same subject (across sessions, viewpoints and illumination conditions) as belonging to the same equivalence class set. The weakly supervised methods (OJ, OT, OE, EX) on the other hand, had only access to the set of orbits formed by partitioning the embedding set in  $129 \times 20 \times 3$  orbits, each

corresponding to all 13 viewpoints for a single identity, illumination condition and session (Fig. 2, row 2).

For the purpose of performance assessment, we used the learned maps  $\Phi$  and encoded the held-out test set (one session). Figure 7 shows the relative distance landscape of the learned embeddings, as 2D t-SNE plots, for all images from 10 subjects of the test set. The weakly supervised OJ appears to have similar or better grouping and separability properties than the fully supervised ST. We used two distance-based tasks to quantitatively evaluate the embedding metric spaces: a *same-different* face verification task and a face retrieval task. In a transformation-robust metric space for face representation, same-identity images should be closer to each other than to other identities, and the nearest neighbor to each should be an image of the same class. We measure the Area Under the ROC Curve (AUC) for verification and the mean top-1 precision for retrieval. The process of training and evaluating an embedding was repeated on all four possible 3-1 splits, across sessions of Multi-PIE, to assess the uncertainty in the performance measures.

For the verification task, we used all unique pairwise distances in the embedding space, considered all possible decision thresholds and integrated the True Positive and False Positive rates to compute the AUC. For the retrieval task, we select the closest point to a query image (top-1 retrieval) from a target search set. We considered each test image individually as query, using the rest of the test set as the target set, after removing all same-identity images (32 in total, including the query) at the same illumination (regardless of viewpoint) and at the same viewpoint (regardless of illumination). This made for a more challenging task and helped in ensuring that the embeddings are evaluated with respect to their preference of identity over appearance, e.g. by excluding candidates with strong pose or illumination bias. As a performance measure, we report the mean precision, i.e. the fraction of queries that yielded a correct retrieval.

Verification performance is shown in Fig. 9 as mean and s.d. of AUC across 3-1 splits of Multi-PIE sessions (3 for embedding training–1 for evaluation). As expected, the weakly supervised methods, that access only the orbit assignments for training, are in-between the ST loss, which has access to category-level labels and the unsupervised AE loss. Orbit triplet (OT) learns very quickly and its performance tends to decrease after a few iterations. The other methods learn at comparable rates. The joint orbit loss (OJ) achieves the best AUC score. A similar ranking holds for the retrieval task, shown as top-1 precision in Fig. 10.

#### 5.4. Early stopping by cross validation

To evaluate the generalization performance, i.e. testing on a set of unseen examples, we trained all embeddings by applying cross-validation for individually selecting the number of iterations. We compared the proposed OJ to the

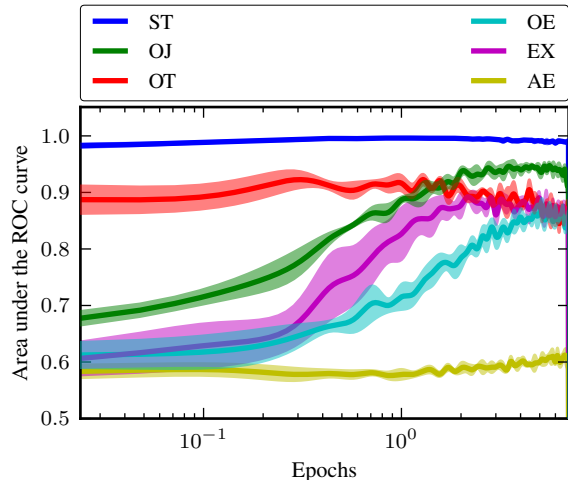


Figure 9: Verification performance on Multi-PIE with epochs of embedding training: Mean Area Under the ROC Curve, with 1 s.d. error bars, over 4 different splits (3 sessions for embedding training, 1 for evaluation).

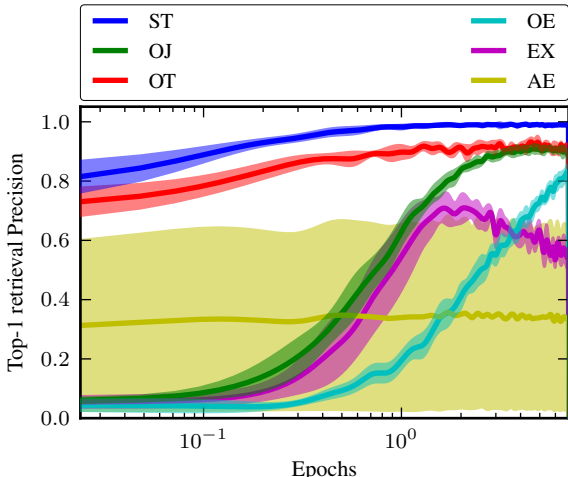


Figure 10: Retrieval performance on Multi-PIE with epochs of embedding training: Mean top-1 precision, with 1 s.d. error bars, over 4 different splits (3 sessions for embedding training, 1 for evaluation).

state-of-the-art, weakly-supervised EX loss using six splits (4 choose 2) by session on Multi-PIE (2-Embedding – 1-Validation (VA) – 1-Test (TE)) and 10 random splits of the MNIST test set (VA – TE), with elements of the same orbit appearing in only one. We selected the stopping time that gave the best performance measure (AUC, top-1 precision, accuracy) on the VA set and evaluated the same measure on the TE set. Table 1 shows the mean and s.d. over splits, with corresponding p-values (paired t-test with Bonferroni correction) quantifying significance for the difference between

	OJ	EX [10]	<i>p</i> -val	OT	<i>p</i> -val	OE	<i>p</i> -val
Multi-PIE AUC	0.94±0.012	0.90±0.007	<b>6e−03</b>	0.93±0.008	6e−01	0.87±0.026	<b>4e−02</b>
Top-1	0.95±0.029	0.80±0.035	<b>2e−02</b>	0.97±0.004	8e−01	0.99±0.002	8e−02
Affine MNIST ACC	0.66±0.027	0.45±0.019	<b>2e−09</b>	0.35±0.021	<b>2e−09</b>	0.49±0.011	<b>6e−08</b>

Table 1: Generalization using embedding-validation-test splits and independent selection of the training stopping time using the validation set. *P*-values quantify significance of the difference between OJ-EX, OJ-OT, and OJ-OE (bolded if  $p < 0.05$ ).

OJ-EX, OJ-OT and OJ-OE. OJ consistently outperforms EX in all three generalization tasks. Moreover, the performance of OJ is either better or statistically indistinguishable (with a standard significance threshold at  $p < 0.05$ ) from OT and OE. This observation makes the case for the joint loss, which can result in substantial improvements like in the one-shot classification task on affine MNIST. Furthermore, it suggests that careful selection of the relative weights ( $\lambda_1$  and  $\lambda_2$ ) of the triplet and reconstruction terms in the OJ loss (Eq. 12), e.g. via cross-validation, could be beneficial.

## 6. Conclusions

We introduced a loss function that combines a *discriminative* and a *generative* term for learning embeddings, using weak supervision from generic transformation orbits. We showed that the resulting image embeddings induce a metric space that is relevant for distance-based learning tasks such as one-shot learning classification, face verification and retrieval. The two loss terms serve complementary purposes, so that joint training is advantageous and supersedes state-of-the-art, exemplar-based training and, when applicable Spatial Transformer Networks. Transformations that do not alter the semantic category of the input are present in most classical perception problems, from pitch shifts in speech recognition to pose, illumination and gait changes in action recognition to reflectance properties for object categorization. The work presented here suggests that explicitly defining equivalence classes according to these transformations is a rich, weak supervision signal that can be exploited in a more general class of representation learning methods, starting from the proposed loss function, to learn semantically relevant embeddings. Such embeddings define distance functions that are robust to typical transformations and are useful for categorization, retrieval, verification and clustering. Future work should assess the relevance for other modalities, such as video or audio and the potential of acquiring the equivalence classes through time continuity.

## Acknowledgements

This material is based upon work supported by the Center for Brains, Minds and Machines (CBMM), funded by NSF STC award CCF-1231216. The DGX-1 used for this research was donated by the NVIDIA Corporation. Stephen Voinea acknowledges

the support of a Nuance Foundation Grant. The authors gratefully acknowledge Tomaso Poggio and Fabio Anselmi for insightful discussions.

## References

- [1] F. Anselmi, J. Z. Leibo, L. Rosasco, J. Mutch, A. Tacchetti, and T. Poggio. Unsupervised Learning of Invariant Representations. *Theoretical Computer Science*, 633:112–121, Jun 2015. 1, 2
- [2] F. Anselmi, L. Rosasco, and T. Poggio. On Invariance and Selectivity in Representation Learning. *Information and Inference*, 5(2):134–158, May 2016. 2, 3, 4
- [3] P. Baldi. Autoencoders, unsupervised learning, and deep architectures. *ICML 2012 Unsupervised and Transfer Learning Workshop*, 2012. 2
- [4] Y. Bengio, A. Courville, and P. Vincent. Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013. 1, 2, 3
- [5] J. Bruna and S. Mallat. Invariant Scattering Convolution Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1872–1886, 2013. 1, 2
- [6] S. Chopra, R. Hadsell, and Y. LeCun. Learning a Similarity Metric Discriminatively, with Application to Face Verification. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 539–546, 2005. 2, 3, 5
- [7] T. S. Cohen and M. Welling. Group Equivariant Convolutional Networks. In *International Conference on Machine Learning (ICML)*, Feb 2016. 1, 3
- [8] J. J. DiCarlo, D. Zoccolan, and N. C. Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–34, feb 2012. 1
- [9] J. Dong and S. Soatto. Domain-Size Pooling in Local Descriptors: DSP-SIFT. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5097–5106, 2015. 2
- [10] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox. Discriminative Unsupervised Feature Learning with Exemplar Convolutional Neural Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(9):1734–1747, Sep 2016. 2, 4, 5, 6, 8, 10
- [11] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-PIE. *Image and Vision Computing*, 28(5):807–813, May 2010. 5, 8
- [12] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality Reduction by Learning an Invariant Mapping. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 1735–1742, 2006. 1, 2

- [13] G. E. Hinton, A. Krizhevsky, and S. D. Wang. Transforming Auto-encoders. In *Artificial Neural Networks and Machine Learning (ICANN 2011)*, volume 6791 of *Lecture Notes in Computer Science*, pages 44–51, 2011. 1, 2
- [14] G. E. Hinton and R. R. Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786):504–7, Jul 2006. 2
- [15] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning (ICML)*, pages 448–456, 2015. 6, 7
- [16] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu. Spatial Transformer Networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2017–2025, 2015. 2, 4, 6, 8
- [17] B. Kulis. Metric Learning: A Survey. *Foundations and Trends® in Machine Learning*, 5(4):287–364, 2013. 3
- [18] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015. 1
- [19] J. Z. Leibo, Q. Liao, F. Anselmi, and T. Poggio. The Invariance Hypothesis Implies Domain-Specific Regions in Visual Cortex. *PLOS Computational Biology*, 11(10):e1004390, oct 2015. 1
- [20] K. Lenc and A. Vedaldi. Understanding Image Representations by Measuring Their Equivariance and Equivalence. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 991–999, 2015. 1
- [21] Q. Liao, J. Z. Leibo, and T. A. Poggio. Learning invariant representations and applications to face verification. In *Advances in Neural Information Processing Systems (NIPS)*, 2013. 2
- [22] S. Mallat. Group Invariant Scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, oct 2012. 2, 3
- [23] J. Masci, U. Meier, D. Cirean, and J. Schmidhuber. Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction. In *21st International Conference on Artificial Neural Networks (ICANN)*, pages 52–59, 2011. 2, 3
- [24] R. Memisevic. Learning to relate images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1829–46, Aug 2013. 2
- [25] H. Mobahi, R. Collobert, and J. Weston. Deep Learning from Temporal Coherence in Video. In *International Conference on Machine Learning (ICML)*, pages 737–744, Jun 2009. 2
- [26] T. Poggio and F. Anselmi. *Visual Cortex and Deep networks: Learning Invariant Representations*. MIT Press, 2016. 1
- [27] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio. Contractive Auto-encoders: Explicit Invariance during Feature Extraction. In *International Conference on Machine Learning (ICML)*, 2011. 2, 3
- [28] M. Sajjadi, M. Javanmardi, and T. Tasdizen. Regularization With Stochastic Transformations and Perturbations for Deep Semi-Supervised Learning. In *Advances in Neural Information Processing System (NIPS)*, 2016. 2
- [29] F. Schroff, D. Kalenichenko, and J. Philbin. FaceNet: A Unified Embedding for Face Recognition and Clustering. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, 2015. 2, 3, 6, 7
- [30] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations (ICLR)*, sep 2015. 6
- [31] S. Soatto and A. Chiuso. Visual Representations: Defining Properties and Deep Approximations. In *International Conference on Learning Representations (ICLR)*, 2016. 1, 2
- [32] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese. Deep Metric Learning via Lifted Structured Feature Embedding. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 4004–4012, 2016. 2, 3
- [33] A. Tacchetti, L. Isik, and T. Poggio. Spatio-temporal neural networks explain human neural representations of action recognition. *arXiv preprint arXiv:1606.04698*, 2016. 2
- [34] L. van der Maaten and G. Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(11):2579–2605, 2008. 7
- [35] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and Composing Robust Features with Denoising Autoencoders. In *International Conference on Machine Learning (ICML)*, pages 1096–1103, Jul 2008. 2, 3
- [36] S. Voinea, C. Zhang, G. Evangelopoulos, L. Rosasco, and T. A. Poggio. Word-level invariant representations from acoustic waveforms. In *INTERSPEECH-2014, 15th Annual Conference of the ISCA*, pages 2385–2389, 2014. 2
- [37] X. Wang and A. Gupta. Unsupervised Learning of Visual Representations Using Videos. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2794–2802, 2015. 2, 5
- [38] K. Q. Weinberger and L. K. Saul. Distance Metric Learning for Large Margin Nearest Neighbor Classification. *The Journal of Machine Learning Research*, 10:207–244, Dec 2009. 2, 3
- [39] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance Metric Learning, with Application to Clustering with Side-information. In *Advances in neural information processing systems (NIPS)*, 2003. 2, 3
- [40] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus. Deconvolutional Networks. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2528–2535, Jun 2010. 3, 6, 7
- [41] C. Zhang, S. Voinea, G. Evangelopoulos, L. Rosasco, and T. Poggio. Discriminative template learning in group-convolutional networks for invariant speech representations. In *INTERSPEECH-2015, 16th Annual Conference of the ISCA*, Dresden, Germany, 2015. 2
- [42] J. Zhao, M. Mathieu, R. Goroshin, and Y. LeCun. Stacked What-Where Auto-encoders. In *International Conference on Learning Representations (ICLR)*, Jun 2016. 2, 3
- [43] W. Zou, S. Zhu, K. Yu, and A. Ng. Deep learning of invariant features via simulated fixations in video. In *Advances in Neural Information Processing System (NIPS)*, 2012. 2, 5

## A. Appendix: Decoder output examples

Figure 3 and Fig. 8 showed examples of rectifications applied on  $g_i x_c$ , where  $x_c$  is a seed, untransformed image, and  $g_i$  an latent transformation, using the output of the decoder  $\tilde{\Phi} \circ \Phi(g_i x)$ . While for OE, this rectification is the sole criterion that drives the learned embedding, for OJ, this competes with the triplet loss term. This will result in different learned features, and image outputs from the decoder.

The effect of the joint training in the visual appearance of the rectified outputs is shown for ten images from affine MNIST dataset in Fig. 11 and ten images from Multi-PIE in Fig. 12. Each one depicts the decoder output for standard AE (column 3), OE (column 4) and OJ (column 5). The autoencoder output is included as a sanity check of the reconstruction loss, i.e., the output is a faithful reconstruction of the input, which includes the transformation effect. The decoders for the OE and OJ losses both do well at rectifying the transformation (affine for MNIST and 3D viewpoint for Multi-PIE), i.e. mapping the transformed input (column 2) to an image that resembles the untransformed one (column 1). One can note subtle differences in the outputs, e.g., the 3 and 9 instances in MNIST but particularly in Multi-PIE, though it is not easy to select one based on visual qualities.



Figure 11: **Column 1**: canonical image, chosen randomly from the MNIST test set. **Column 2**: seed image generated from the canonical via a random affine transformation. **Columns 3-5** show the result of passing the seed image through the decoders of the Autoencoder, Orbit Encode and Orbit Joint trained networks respectively.



Figure 12: **Column 1**: canonical image, chosen randomly from the Multi-PIE test set. **Column 2**: seed image generated from the canonical via a random affine transformation. **Columns 3-5** show the result of passing the seed image through the decoders of the Autoencoder, Orbit Encode and Orbit Joint trained networks respectively.