This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the version available on IEEE Xplore.

Deep Unsupervised Similarity Learning using Partially Ordered Sets

Miguel A. Bautista*, Artsiom Sanakoyeu*, Björn Ommer
Heidelberg Collaboratory for Image Processing
IWR, Heidelberg University, Germany
firstname.lastname@iwr.uni-heidelberg.de

Abstract

Unsupervised learning of visual similarities is of paramount importance to computer vision, particularly due to lacking training data for fine-grained similarities. Deep learning of similarities is often based on relationships between pairs or triplets of samples. Many of these relations are unreliable and mutually contradicting, implying inconsistencies when trained without supervision information that relates different tuples or triplets to each other. To overcome this problem, we use local estimates of reliable (dis-)similarities to initially group samples into compact surrogate classes and use local partial orders of samples to classes to link classes to each other. Similarity learning is then formulated as a partial ordering task with soft correspondences of all samples to classes. Adopting a strategy of self-supervision, a CNN is trained to optimally represent samples in a mutually consistent manner while updating the classes. The similarity learning and grouping procedure are integrated in a single model and optimized jointly. The proposed unsupervised approach shows competitive performance on detailed pose estimation and object classification.

1. Introduction

Visual similarities lie at the heart of a large number of computer vision tasks ranging from low-level image processing to high-level understanding of human poses or object classification. Of the numerous techniques for similarity learning, supervised methods have been a popular technique, leading to formulations in which similarity learning was casted as a ranking [36], regression [8], and classification [23] task. In recent years, with the advent of Convolutional Neural Networks (CNN), formulations based on a ranking (i.e. ordering) of pairs or triplets of samples according to their similarity have shown impressive results [33]. However, to achieve this performance boost, these CNN architectures require millions of samples of supervised training data or at least the fine-tuning [5] on large datasets such as PASCAL VOC.

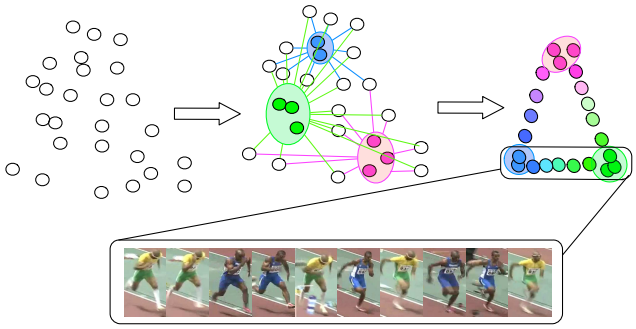

Figure 1. Visualization of the interaction between surrogate classes and partially ordered sets (posets). Our approach starts with a set of unlabeled samples, building small surrogate classes and generating posets to unlabeled samples to learn fine-grained similarities.

Although the amount of accessible image data is growing at an ever increasing rate, supervised labeling of similarities is very costly. In addition, not only similarities between images are important, but especially between objects and their parts. Annotating the fine-grained similarities between all these entities is a futile undertaking, in particular for the large-scale datasets typically used for training CNNs. Deep unsupervised learning of similarities is, therefore, of great interest to the vision community, since it does not require any labels for pre-training or fine-tuning. In this way we can utilize large image datasets without being limited by the need for costly manual annotations.

To utilize the vast amounts of available unlabeled training data, there is a quest to leverage context information intrinsic to images/video for *self*-supervision. However, this context is typically highly local (i.e position of patches in the same image [5], object tracks through short number of frames [33] or image impainting [22]), establishing relations between tuples [5] or triplets [20, 38, 33] of images. Hence, these approaches utilize loss functions that order a positive I_p and a negative I_n image with respect to an anchor image I_a so that, $d(I_a, I_p) < d(I_a, I_n)$. During train-

*Both authors contributed equally to this work.

ing, these methods rely on the CNN to indirectly learn comparisons between samples that were processed in independent training batches, and generalize to unseen data.

Instead of relying on the CNN to indirectly balance and learn sample comparisons unseen during training, a more natural approach is to explicitly encode richer relationships between samples as supervision. In this sense, an effective approach to tackle unsupervised similarity learning is to frame it as a series of surrogate (i.e. artificially created) classification tasks [6, 3]. Therefore, mutually similar samples are assigned the same class label, otherwise a different label. To obtain surrogate classification tasks, compact groups of mutually similar samples are computed by clustering [3] over a weak initial representation (e.g standard features such as HOG). Then, each group receives a mutually exclusive label and a CNN is trained to solve the associated classification problem, thereby learning a representation that encodes similarity in the intermediate layers. However, given the unreliability of initial similarities, a large number of training samples are neither mutually similar nor dissimilar and are, thus, not assigned to any of the compact surrogate classes. Consequentially they are ignored during training, hence overlooking important information. Also, classification can yield fairly coarse similarities, considering the discrete nature of the classes. Furthermore, the similarities learnt by the different classification tasks are not optimized jointly, which can lead to mutually contradicting relationships, since transitivity is not captured.

To overcome the fundamental limitations of these approaches we propose to: *(i)* Cast similarity learning as a surrogate classification task, using compact groups of mutually related samples as surrogates classes in a self-supervision spirit. *(ii)* Combine classification with a partial ordering of samples. Even samples, which cannot be assigned to any surrogate class due to unreliable initial similarities are thus incorporated during training and in contrast to discrete classification, more fine-grained relationships are obtained due to the ordering. *(iii)* Explicitly optimize similarities in a given representation space, instead of using the representation space indirectly learnt by intermediate layers of a CNN trained for classification.*(iv)* Jointly optimize the surrogate classification tasks for similarity learning and the underlying grouping in a recurrent framework which is end-to-end trainable. Fig. 2 shows a conceptual pipeline of the proposed approach.

Experimental evaluation on diverse tasks of pose estimation and object classification shows state-of-the-art performance on standard benchmarks, thus underlining the wide applicability of the proposed approach. In the pose estimation experiments we show that our method learns a general representation, which can be transferred across datasets and is even valuable for initialization of supervised methods. In addition, in the object classification experiments we successfully leverage large unlabeled datasets to learn representations in the fashion of zero-shot learning.

2. Related Work

Similarity learning has been a problem of major interest for the vision community from its early beginnings, due to its broad applications. With the advent of CNNs, several approaches have been proposed for supervised similarity learning using either pairs [39], or triplets [32] of images. Furthermore, recent works by Misra et al. [20], Wang et al. [33], and Doersh et al. [5] showed that temporal information in videos and spatial context information in images can be utilized as a convenient supervisory signal for learning feature representation with CNNs in an unsupervised manner. However, either supervised or unsupervised, all these formulations for learning similarities require that the supervisory information scales quadratically for pairs of images, or cubically for triplets. This results in very large training time. Furthermore, tuple and triplet formulations advocate on the CNN to indirectly learn to conceal unrelated pairs of samples (i.e. pairs that were not tied to any anchor) that are processed in different, independent batches during training. Another recent approach that has been proposed for learning similarities in an unsupervised manner is to build a surrogate (i.e. an artificial) classification task either by utilizing heavy data augmentation [6] or by clustering based on initial weak estimates of similarities [3, 15]. The advantage of these approaches over tuple or triplet formulations is that several relationships of similarity (samples in the same class) and dissimilarity (samples in other classes) between samples are utilized during training. This results in more efficient training procedures, avoiding to sample millions of pairs or triplets of samples and encoding richer relationships between samples.

In addition, similarity learning has also been studied from the perspective of metric learning approaches [35, 26, 25]. In the realm of supervised metric learning methods, Roweis et. al [26] formulated metric learning as a cross-entropy based classification problem in which all pairwise neighbouring samples are pulled together while non-neighbouring samples are pushed away. However, provided that clusters of neighbouring points can have an arbitrary large number of samples, this strategy fails to scale to the large image collections used for unsupervised learning of similarities. Further efforts [28, 19] have tried to reduce the computational cost of performing all pairwise comparisons [17]. Recently, [34] leveraged low-density classifiers to enable the use of large volumes of unlabelled data during training. However, [34] cannot be successfully applied to the unsupervised scenario, since it requires a strongly supervised initialization , e.g. an ImageNet pre-trained model.

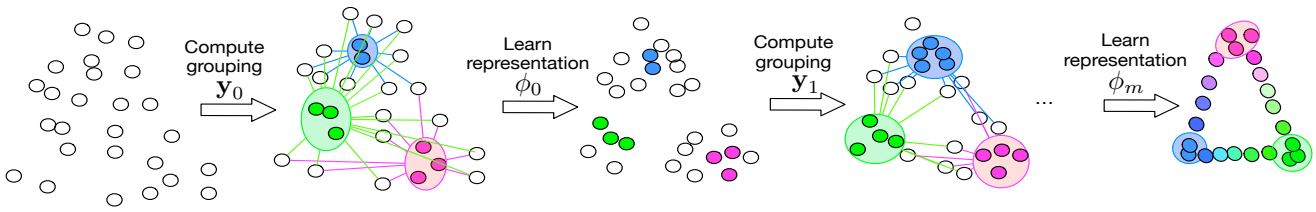

Figure 2. Visual summary of our approach. In the $\mathbf{y}$-steps the clustering procedure computes surrogate classes (shaded in color) based on the current representation. In the ϕ-steps we learn a representation using the surrogate classes and partial orders of samples not assigned to any surrogate class (samples in white), by pulling them closer to their nearest classes and pushing them further from the rest.

3. Approach

In this section we show how to combine partially ordered sets (posets) of samples and surrogate classification to learn fine-grained similarities in an unsupervised manner. Key steps of the approach include: *(i)* Compute compact groups of mutually related samples and use each group as a surrogate class in a classification task. *(ii)* Learn fine-grained similarities by modelling partial orderings to also leverage those samples that cannot be assigned to a surrogate class. *(iii)* Due to the interdependence of grouping and similarity learning we jointly optimize them in a recurrent framework. Fig. 2 shows a visual example of the main steps of our approach.

3.1. Grouping

To formulate unsupervised similarity learning as a classification approach we need to define surrogate classes, since labels are not available. To compute these surrogate classes we first gather compact groups of samples using standard feature distances (LDA whitened HOG [12, 27, 7]). HOG-LDA is a computationally effective foundation for estimating similarities between a large number of samples. Let our training set be defined as $\mathbf{X} \in \mathbb{R}^{n \times p}$, where n is the total number of samples and $\mathbf{x}_i$ is the $i-$th sample. Then, the HOG-LDA similarity between a pair of samples $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined as $s_{ij} = \exp(-\|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|_2)$. Here $\phi(\mathbf{x}_i) \in \mathbb{R}^{1 \times d}$ is the $d-$dimensional representation of sample $\mathbf{x}_i$ in the HOG-LDA feature space.

Albeit unreliable to relate all samples to another, HOG-LDA similarities can be used to find the nearest and furthest neighbors, as highly similar and dissimilar samples to a given anchor sample $\mathbf{x}_i$ stand out from the similarity distribution. Therefore, to build surrogate classes (i.e. compact groups of samples) we group each $\mathbf{x}_i$ with its immediate neighborhood (samples with similarity within the top 5%) so that all merged samples are mutually similar. These groups are compact, differ in size, and may be mutually overlapping. To reduce redundancy, highly overlapping classes are subsequently merged by agglomerative clustering, which terminates if intra-class similarity of a surrogate class is less than half of its constituents. We denote the set of samples assigned to the c-th surrogate class as $\mathcal{C}_c$, and the label assigned to each sample as $\mathbf{y} \in \{-1, 0, \dots, C-1\}^{1 \times n}$, where the label assigned to sample $\mathbf{x}_i$ is denoted as y_i. All samples that are not assigned to any surrogate class get label -1.

3.2. Partially Ordered Sets

Provided the unreliability of similarity estimates used for building surrogate classes, a large number of samples cannot be assigned to any class, because they are neither similar nor dissimilar to any sample. This deprives the optimization of using all available data during training. As a result, fine-grained similarities are poorly represented, since learning to classify surrogate classes does not model relative similarities of samples that are not assigned to any class. To overcome this limitation we leverage the information encoded in posets of samples relative to a surrogate class. That is, for each sample not assigned to any surrogate class (i.e. $\mathbf{x}_i : y_i = -1$) we compute a soft assignment (i.e. a similarity score) to the Z nearest surrogate classes $\mathcal{C}_z : z \in \{1, \dots, Z\}$. Once all unlabeled points are softly assigned to their Z nearest classes, we obtain as a result, a poset $\mathcal{P}_c$ for each class. Thus, a poset $\mathcal{P}_c$ is a set of samples which are softly assigned to class $\mathcal{C}_c$. Posets can be of variable size and partially overlapping. We show a visual example of a poset in Fig. 3.

Formally, given a deep feature representation ϕ^θ (e.g an arbitrary layer in a CNN with parameters θ), and a surrogate class $\mathcal{C}_c$, a poset of unlabeled samples $\mathcal{P}_c = \{\mathbf{x}_j, \dots, \mathbf{x}_k\} : y_j = y_k = -1 \; \forall \; j, k$ with respect to $\mathcal{C}_c$ is defined as:

$$\forall_{\mathbf{x}_i \in \mathcal{C}_c} \{\exp(-\|\phi^\theta(\mathbf{x}_i) - \phi^\theta(\mathbf{x}_j)\|_2) > \exp(-\|\phi^\theta(\mathbf{x}_i) - \phi^\theta(\mathbf{x}_k)\|_2)\} \iff j < k \forall j, k. \quad (1)$$

In Eq. (1) a poset is defined by computing the similarity of unlabeled sample $\mathbf{x}_j$ to all the samples in class $\mathcal{C}_c$, which during training is costly to optimize. However, due the compactness of our grouping approach, which only gathers very similar samples into surrogate $\mathcal{C}_c$, we can effectively replace the similarities to all points in $\mathcal{C}_c$ by the similarity to a representative sample $\bar{\mathbf{x}}_c$ in $\mathcal{C}_c$, which is the class medioid, $\bar{\mathbf{x}}_c = \underset{\mathbf{x}_i \in \mathcal{C}_c}{\operatorname{argmin}} \sum_{\mathbf{x}_j \in \mathcal{C}_c} \|\phi^\theta(\mathbf{x}_i) - \phi^\theta(\mathbf{x}_j)\|_2$.

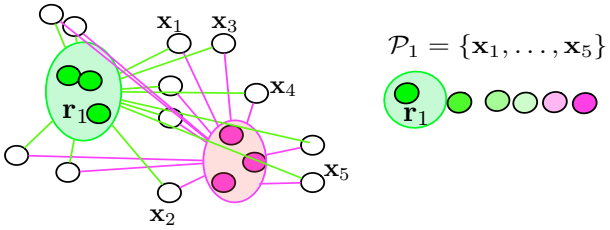

Figure 3. Visual interpretation of a poset. Samples assigned to a surrogate class are shaded in a particular color, while samples not assigned to surrogate classes are represented in white.

Following the definition of a poset in Eq. 1, the widely adopted tuple and triplet formulations [5, 33, 20, 38] are a specific case of a poset in which $\mathcal{P}$ contains at most 2 samples, and $\mathcal{C}_c$ contains just one. In this sense, deep feature representations ϕ (i.e. CNNs) trained using triplet losses seek to sort two pairs of samples (i.e. anchor-positive and anchor-negative) according to their similarity. As a result, triplet formulations rely on the CNN to *indirectly* learn to compare and reconcile the vast number of *unrelated* sampled pairs that were processed on different, independent mini-batches during training. In contrast, posets, explicitly encode an ordering between a large number of sample pairs (i.e pairs consisting of an unlabeled sample and its nearest class representative). Therefore, using posets during training enforces the CNN to order all unlabeled samples $\mathbf{x}_i : y_i = -1$ according to their similarity to the Z nearest class representatives $\mathbf{r}_i^z : z \in \{1, \ldots, Z\}$, where $\mathbf{r}_i^z$ is the $z-$th nearest $\bar{\mathbf{x}}_c$ to sample $\mathbf{x}_i$, learning fine-grained interactions between samples. Posets generalize tuple and triplet formulations by encoding similarity relationships between unlabeled samples to make a decision whether to move closer to a surrogate class. This effectively increases our training set when compared to just using the samples assigned to surrogate classes, and allows us to model finer relationships.

3.3. Objective function

In our formulation, we strive for a trade-off model in which we jointly optimize a surrogate classification task and a metric loss to capture the fine-grained similarities encoded in posets. Therefore, we seek an objective function $\mathcal{L}$ which penalizes: *(i)* misclassifications of samples $\mathbf{x}_i$ with respect to their surrogate label y_i, and *(ii)* similarities of samples $\mathbf{x}_i : y_i = -1$. with respect to their Z nearest class representatives. The objective function should inherit the reliability of framing similarity learning as surrogate classification tasks, while using posets to incorporate those training samples that were previously ignored because they could not be assigned to any surrogate class. In particular, we require the CNN to pull samples from posets $\mathbf{x}_i \in \mathcal{P}_c$ closer to their Z nearest class representatives, while pushing them further from all other class representatives in a training mini-batch. Furthermore, we require that unreliable similarities (i.e. samples that are far from all surrogate classes), vanish from the loss, rendering the learning process robust to outliers. In addition, in order to capture fine-grained similarity relationships, we want to directly optimize the feature space ϕ in which similarities are computed.

Therefore, let $\mathbf{R}^z \in \mathbb{R}^{n \times d}$ denote the z-th nearest class representatives of each unlabeled sample $\mathbf{x}_i : y_i = -1$, where $\mathbf{r}_i^z$ is the z-th nearest class representative of sample $\mathbf{x}_i$, and θ be the parameters of the CNN. Then, our objective function combines the surrogate classification loss $\mathcal{L}_1$ with our poset loss $\mathcal{L}_2$:

$$\mathcal{L}(\mathbf{x}_i, y_i, \mathbf{R}; \theta) = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_1(\mathbf{x}_i, y_i) + \lambda \mathcal{L}_2(\mathbf{x}_i, \mathbf{R}, \phi), \tag{2}$$

where λ is a scalar and,

$$\mathcal{L}_1(\mathbf{x}_i, y_i; \theta) = -\log \frac{\exp(t^\theta_{i,y_i})}{\sum_{j=0}^{C-1} \exp(t^\theta_{i,j})} \mathbb{1}_{y_i \neq -1}, \tag{3}$$

$$\mathcal{L}_2(\mathbf{x}_i, \mathbf{R}; \theta) = \\ = -\log \frac{\sum_{z=1}^{Z} \exp(\frac{-1}{2\sigma^2}(\| \phi^\theta(\mathbf{x}_i) - \phi^\theta(\mathbf{r}_i^z)\|_2^2 - \gamma))}{\sum_{j=1}^{C'} \exp(\frac{-1}{2\sigma^2} \| \phi^\theta(\mathbf{x}_i) - \phi^\theta(\mathbf{r}_j)\|_2^2)}. \tag{4}$$

In Eq. (3), $\mathbf{t}_i^\theta = \mathbf{t}^\theta(\mathbf{x}_i)$ are the logits of sample $\mathbf{x}_i$ for a CNN with parameters θ. In Eq. (4) C' is the number of surrogate classes in the batch, σ is the standard deviation of the current assignment of samples to surrogate classes, and γ is the margin between surrogate classes. It is note-worthy that Eq. (4) can scale to an arbitrary number of classes, since it does not depend on a fixed-sized output target layer, avoiding the shortcomings of large output spaces in CNN learning [31] [1].

Finally, note that if $Z = 1$ the problem reduces to a cross-entropy based classification, where the standard logits (i.e. outputs of the last layer) are replaced by the similarity to the surrogate class representative in feature space ϕ. However, for $Z > 1$ relative similarities between surrogate classes enter into play and posets encoding fine-grained interactions naturally arise (cf. Fig. 5). In all our experiments we set $Z >= 2$. During training, CNN parameters θ are updated by error-backpropagation with stochastic mini-batch gradient descent. In typical classification scenarios the training set is randomly shuffled to avoid biased gradient computations that hamper the learning process. Therefore, at training time we build our mini-batches of samples by selecting a random set of samples not assigned to a surrogate

[1] In our experiments we successfully scaled the output space to 20K surrogate classes.

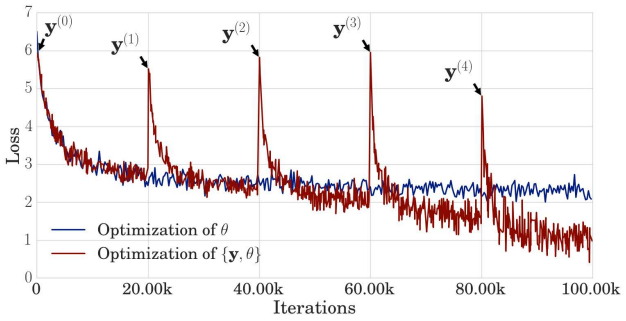

Figure 4. Loss value $\mathcal{L}$ for long jump category over each unrolling step. Evidently the model benefits from jointly optimizing $\{\mathbf{y}, \theta\}$.

class $\mathbf{x}_i : y_i = -1$, and retrieving all the surrogate classes $\mathcal{C}_c$ which contain $\mathbf{x}_i$ in their poset $\mathbf{x}_i \in \mathcal{P}_c$. In Fig. 4 we take as a study case the *long jump* category of the Olympic Sports dataset (cf. Sec. 4) and show the $\mathcal{L}$ decreases along iterations. In particular, we show that if $\mathbf{y}$ and θ are optimized jointly we attain better performance.

3.4. Joint Optimization

In our setup, the grouping and similarity learning tasks are mutually dependent on each other. Therefore, we strive to jointly learn a representation ϕ^θ, which captures similarity relationships, and an assignment of samples to surrogate classes $\mathbf{y}$. A natural way to model such dependence in variables is to use a Recurrent Neural Network (RNN) [18]. In particular, RNNs have shown a great potential to model relationships on sequential problems, where each prediction depends on previous observations. Inspired by this insight, we employ a recurrent optimization technique. Following the standard process for learning RNNs we jointly learn $\{\mathbf{y}, \theta\}$ by unrolling the optimization into steps. At time step m we update $\mathbf{y}$ and θ as follows:

$$\begin{aligned} \mathbf{y}^{(m)} &= \operatorname*{argmax}_{\mathbf{y}} \mathcal{G}(\mathbf{X}; \phi^{\theta^{(m-1)}}, \mathbf{y}^{(m-1)}) \\ \text{s.t.} \sum_{i:y_i=c}^{n} 1 &> t, \forall_{c\in\{0,\dots,C-1\}}, \end{aligned} \tag{5}$$

$$\theta^{(m)} = \operatorname*{argmin}_{\theta} \mathcal{L}(\mathbf{X}, \mathbf{y}^{(m)}, \mathbf{R}^{(m)}; \theta^{(m-1)}). \tag{6}$$

Where $\mathcal{G}$ is a cost function of pairwise clustering that favors compactness based on sample similarities, which are entailed by the representation $\phi^{\theta^{(m-1)}}$, and t is a lower bound on the number of samples of each cluster.

$$\mathcal{G}(\mathbf{X}; \phi^\theta, \mathbf{y}) = \\ = \sum_{c=0}^{C-1} \sum_{i:y_i=c}^{n} \frac{\sum_{j:y_j=c}^{n} \exp(-\|\phi^\theta(\mathbf{x}_i) - \phi^\theta(\mathbf{x}_j)\|_2)}{\left(\sum_{j:y_j=c}^{n} 1\right)^2}. \tag{7}$$

In order to avoid the trivial solution of assigning a single sample to each cluster we initialize $\mathbf{y}^{(0)}$ with the grouping introduced in Sec. 3.1 using HOG-LDA as our initial ϕ. In our implementation, $\mathbf{y}$ follows a relaxed one-hot encoding, which can be interpreted as an affinity of samples to clusters. Then, Eq. (5) becomes differentiable and is optimized using SGD. Subsequently, $\mathcal{L}$ learns a deep similarity encoding representation $\phi^{\theta^{(m)}}$ on samples $\mathbf{X}$ using assignments $\mathbf{y}^{(m)}$ and partial orders of $\mathbf{X}$ with respect to representatives $\mathbf{R}^{(m)}$. In a typical RNN scenario, for each training iteration the RNN is unrolled m steps. However, this would be inefficient in our setup, as the CNN representation ϕ^θ is learnt using SGD, and thus, requires to be optimized for a large number of iterations to be reliable, especially at the first unrolled steps. Therefore, at each step m, we find $\theta^{(m)}$ by optimizing Eq. (6) for a number of iterations, fixing $\mathbf{y}^{(m)}$ and $\mathbf{R}^{(m)}$. Then, we use $\theta^{(m)}$ to find the optimal $\mathbf{y}^{(m+1)}$ by optimizing $\mathcal{G}$ using SGD. The presented RNN can also be interpreted as block-coordinate descent [37], where the grouping $\mathbf{y}$ is fixed while updating the representation parameters θ and vice versa. The convergence of block coordinate-descent methods has been largely discussed obtaining guarantees of convergence to a stationary point [30, 4].

4. Experiments

In this section we present a quantitative and qualitative analysis of our poset based approach on the challenging and diverse scenarios of human pose estimation and object classification. In all our experiments we adopt the AlexNet architecture [14].

4.1. Human Pose Estimation

To evaluate the proposed approach in the context of pose estimation we consider 3 different datasets, Olympic Sports (OS), Leeds Sports Pose (LSP), and MPII-Pose (MPI). We show that our unsupervised method is valuable for a range of retrieval problems: For OS we evaluate zero-shot retrieval of detailed postures. On LSP, we perform zero-shot and semi-supervised estimation of pose. Finally, on MPII we evaluate our approach as an initialization for a supervised learning approach for pose estimation. In contrast to other methods that fine-tune supervised initializations of a CNN, we train our AlexNet [14] architecture from scratch.

4.1.1 Olympic Sports

The Olympic Sports dataset [21] is a compilation of video sequences of different 16 sports competitions, containing more than 110000 frames overall. We use the approach of [10] to compute person bounding boxes and utilize this large dataset to learn a general representation that encodes

fine-grained posture similarities. In order to do so, we initially compute 20000 surrogate classes consisting of 8 samples in average. Then, we utilize partially ordered sets of samples not assigned to any surrogate classes. To train our RNN we use the optimization approach described in Sec. 3.4, where the RNN is unrolled on $m = 10$ steps. At each unrolled step, θ is updated during 20000 iterations of error-backpropagation. To evaluate our representation on fine-grained posture retrieval we utilize the annotations provided by [3] on their project webpage [2] and follow their evaluation protocol, using their annotations only for testing. We compare our method with CliqueCNN [3] by directly evaluating their models provided at[2], the triplet formulation of Shuffle&Learn [20], the tuple approach of Doersch et. al [5], Exemplar-CNN [6], Alexnet [14], Exemplar-SVMs [16], and HOG-LDA [12]. For completeness we also include a version of our model that was initialized with Imagenet model [14]. During training we use as ϕ the *fc7* output representation of Alexnet and compute similarities using cosine distance. We use *Tensorflow* [1] for our implementation. (*i*) For CliqueCNN, Shuffle& Learn, and Doersh et. al methods we use the models downloaded from their respective project websites. *(ii)* Exemplar-CNN is trained using the best performing parameters reported in [6] and the 64c5-128c5-256c5-512f architecture. Then we use the output of fc4 and compute 4-quadrant max pooling. *(iii)* Exemplar-SVM was trained on the exemplar frames using the HOG descriptor. The samples for hard negative mining come from all categories except the one that an exemplar is from. We performed cross-validation to find an optimal number of negative mining rounds (less than three). The class weights of the linear SVM were set as $C_1 = 0.5$ and $C_2 = 0.01$. During training of our approach, each image in the training set is augmented by performing random translation, scaling and rotation to improve invariance with respect to these.

In Tab. 1 we show the average AuC over all categories for the different methods. When compared with the best runner up [3], the proposed approach improves the performance 2% (the method in [3] was pre-trained on Imagenet). This improvement is due to the additional relationships established by posets on samples not assigned to any surrogate class, which [3] ignored during training. In addition, when compared to the state-of-the-art methods that leverage tuples [5] or triplets [20] for training a CNN from scratch, our approach shows 16% higher performance. This is explained by the more detailed similarity relationships encoded in each poset, which in tuple methods the CNN has to learn implicitly.

In addition to the quantitative analysis we also perform a qualitative evaluation of the similarities learnt by the proposed method. In order to do so, we take a sequence from

[2] https://asanakoy.github.io/cliquecnn/

HOG-LDA [12]	Ex-SVM [16]	Ex-CNN [6]
0.62	0.72	0.64
Alexnet [14]	Doersch et. al [5]	Suffle&Learn [20]
0.65	0.62	0.63
CliqueCNN [3]	Ours scratch	Ours Imagenet
0.83	0.78	**0.85**

Table 1. Avg. AUC for each method on Olympic Sports dataset.

the *long jump* category of Olympic Sports and select two representatives $\{\mathbf{r}_1, \mathbf{r}_r\}$ with a gap of 8 frames between them and show in Fig. 5 the poset learnt by our approach. The top row shows two representatives of the same sequence highlighted in red and the remaining sub-sequence between them in blue. In the bottom row, we present the poset learnt by our approach. Since $\mathbf{r}_1$ and $\mathbf{r}_2$ show different parts of a short gait cycle, the similarity relations in the poset should set other frames into perspective and order them. And indeed, we observe that the poset successfully encodes this temporal coherence by ordering frames from other sequences that fit in this gap. This is even more interesting, since during training absolutely no temporal structure was introduced in the model, as we were training on only individual frames. These results spurred our interest to also apply the learnt posets for video reconstruction using only few sparse representatives per sequence, additional results can be found in the supplementary material.

4.1.2 Leeds Sports Pose

After evaluating the proposed method for fine-grained posture retrieval, we tackle the problem of zero-shot pose estimation on the LSP dataset. That is, we transfer the pose representation learnt on Olympic Sports to the LSP dataset and retrieve similar poses based on their similarity. The LSP [13] dataset is one of the most widely used benchmarks for pose estimation. In order to evaluate our model we then employ the fine-grained pose representation learnt by our approach on OS, and transfer it to LSP, without doing any further training. For evaluation we use the representation to compute visual similarities and find nearest neighbours to a query frame. Since the evaluation is zero-shot, joint labels are not available. At test time we therefore estimate the joint coordinates of a query person by finding the most similar frame from the training set and taking its joint coordinates. We then compare our method with Alexnet [14] pre-trained on Imagenet, the triplet approach of Misra et. al (Shuffle&Learn) [20] and CliqueCNN [3]. In addition, we also report an upper bound on the performance that can be achieved by zero-shot evaluation using ground-truth similarities. Here the most similar pose for a query is given by the frame, which is closest in average distance of ground-truth pose annotations. This is the best one can achieve without a parametric model for pose (the performance gap to 100% shows the discrepancy between poses in test and

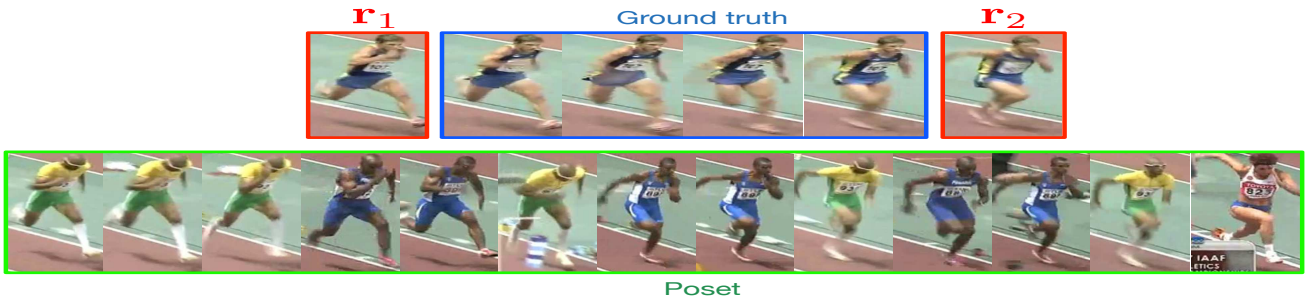

Figure 5. Partially ordered set learnt by the proposed approach. The top row shows two surrogate class representatives (highlighted in red) of the same sequence and the ground truth sub-sequence between them highlighted in blue. The bottom row shows the predicted poset highlighted in green, successfully capturing fine-grained similarities.

Method	T	UL	LL	UA	LA	H	Total
Ours - Imagenet	83.5	54.0	46.8	34.1	16.8	54.3	**48.3**
CliqueCNN [3]	80.1	50.1	45.7	27.2	12.6	45.5	43.5
Alexnet[14]	76.9	47.8	41.8	26.7	11.2	42.4	41.1
Ours - Scratch	67.0	38.6	34.9	20.5	9.8	35.1	**34.3**
Shuffle&Learn [20]	60.4	33.2	28.9	16.8	7.1	33.8	30.0
Ground Truth	93.7	78.8	74.9	58.7	36.4	72.4	69.2
P. Machines [24]	93.1	83.6	76.8	68.1	42.2	85.4	72.0

Table 2. PCP measure for each method on Leeds Sports dataset for zero-shot pose estimation.

Initialization	T	UL	LL	UA	LA	H	Total
Ours	89.7	62.1	48.2	36.0	16.0	54.2	51.0
Shuffle&Learn [20]	90.4	62.7	45.7	33.3	11.8	52.0	49.3
Random init.	87.3	52.3	35.4	25.4	7.6	44.0	42.0
Alexnet [14]	92.8	68.1	53.0	39.8	17.5	62.8	55.7

Table 3. PCP measure for each method on Leeds Sports dataset using different methods as initialization for the DeepPose method [29].

train set). For completeness, we compare with a fully supervised state-of-the-art approach for pose estimation [24]. For computing simialarities we use the same experimental settings described in Sect. 4.1.1, where ϕ is the representation extracted from *pool5* layer of Alexnet. In Tab. 2 we show the PCP@0.5 obtained by the different methods. For a fair comparison with CliqueCNN [3] (which was pre-trained on Imagenet), we include a version of our method trained using Imagenet initialization. Our approach significantly improves the visual similarities learned using both Imagenet pre-trained AlexNet and CliqueCNN [3], obtaining a performance boost of at least 4% in PCP score. In addition, when trained from scratch without any pre-training on Imagenet our model outperforms the recent triplet model of [20] by 4%, due to the fact that posets are a natural generalization of triplet models, which encode finer relationships between samples. Finally, it is notable that even though our pose representation is *transferred from a different dataset* without fine-tuning on LSP, it obtains state-of-the-art performance. In Fig. 6 we show a qualitative comparison of the part predictions of the supervised approach in [29] trained on LSP, with the heatmaps yielded by our zero-shot approach.

In addition to the zero-shot learning experiments we also used our pose representation learnt on Olympic Sports as an initialization for learning the DeepPose method [29] on LSP in a semi-supervised fashion. To evaluate the validity of our representation we compare the performance obtained by DeepPose [29], when trained with one of the following models as initialization: random initialization, Shuffle&Learn [20] (triplet model), and our approach trained on OS. For completeness, we also compared with Imagenet pre-trained AlexNet [14]. Tab. 3 shows the PCP@0.5 obtained by training DeepPose (stg-1) using their best reported parameters. The obtained results show that our representation successfully encodes pose information, obtaining a performance boost of 9% when compared with a random initialization (that our model starts from), since we learn general pose features that act as a regularizer during training. A note-worthy comparison is that the difference between utilizing Imagenet pre-training, which uses 1.2 million labeled images, and our unsupervised learning approach is just 5%.

4.1.3 MPII Pose

We now evaluate our approach in the challenging MPII Pose dataset [2] which is a state of the art benchmark for evaluation of articulated human pose estimation. The dataset includes around 25K images containing over 40K people with annotated body joints. MPII Pose is a particularly challenging dataset because of the clutter, occlusion and number of persons appearing in images. To evaluate our approach in MPII Pose we follow the semi-supervised training protocol used for LSP and compare the performance obtained by DeepPose [29], when trained using as initialization each of the following models: Random initialization, Shuffle&Learn [20] (triplet model) and our approach trained on OS. For completion, we also evaluate Imagenet pre-trained AlexNet [14] as initialization. Following the standard evaluation metric on MPII dataset, Tab. 4 shows the PCKh@0.5 obtained by training DeepPose (stg-1) us-

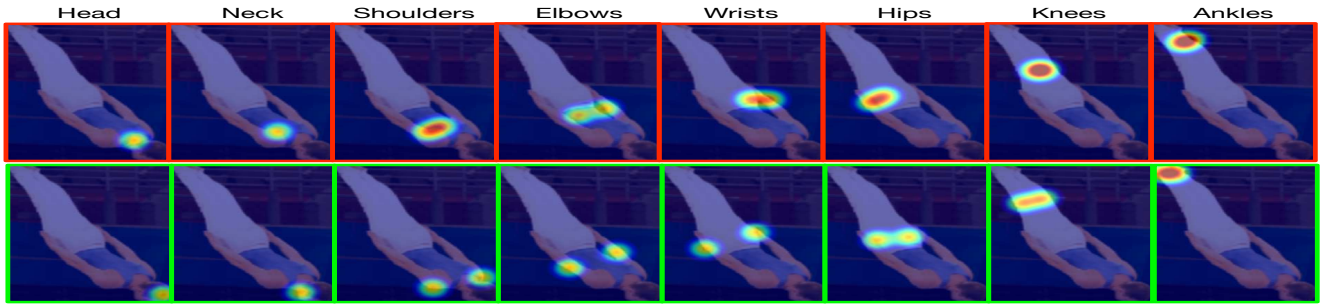

Figure 6. Top row: Heatmaps obtained by DeepPose (stg-1) [29] trained on LSP, highlighted in red. Bottom row: Heatmaps obtained by our zero-shot unsupervised approach, highlighted in green.

	Ours	Shuffle&Learn [20]	Random Init.	AlexNet[14]
Head	83.8	75.8	79.5	87.2
Neck	90.9	86.3	87.1	93.2
LR Shoulder	77.5	75.0	71.6	85.2
LR Elbow.	60.8	59.2	52.1	69.6
LR Wrist	44.4	42.2	34.6	52.0
LR Hip	74.6	73.3	64.1	81.3
LR Knee	65.4	63.1	58.3	69.7
LR Ankle	57.4	51.7	51.2	62.0
Thorax	90.5	87.1	85.5	93.4
Pelvis	81.3	79.5	70.1	86.6
Total	72.7	69.3	65.4	78.0

Table 4. PCKh@0.5 measure for each initialization method on MPII Pose benchmark dataset using different initializations for the DeepPose approach [29].

ing their best reported parameters with the different initializations.

The performance obtained on MPII Pose benchmark shows that our unsupervised representation successfully scales to challenging datasets, successfully dealing with clutter, occlusions and multiple persons. In particular, when comparing our unsupervised initialization with a random initialization we obtain a 7% performance boost, which indicates that our features encode a robust notion of pose that is robust to the clutter present in MPII dataset. Furthermore, we obtain a 3% improvement over the Shuffle&Learn [20] approach, due to the finer-grained relationships encoded by posets. Finally, it is important to note that the difference between utilizing Imagenet pre-trained AlexNet[14], and our unsupervised learning approach is just 5%.

4.2. Object Classification on PASCAL VOC

To evaluate the general applicability of our approach, let us now switch from human pose estimation to the challenging diverse problem of object classification. We classify object bounding boxes of the PASCAL VOC 2007 [9] dataset in zero-shot fashion by predicting the most similar images to a query. The object representation needed for computing similarities, we obtain without supervision information, using visual similarities of the triplet model of Wang et al. [33] as initializiation. Neither this initialization nor our method apply pre-training or fine tuning on ImageNet or Pascal VOC. Using this initialization we then compute an initial clustering on 1000 surrogate classes with 8 samples in average, on the training set images. We then utilize partially ordered sets of samples not assigned to any class, and jointly optimize assignments and representation using the recurrent optimization approach describe in Sec. 3.4. The representation ϕ used to compute similarities on the PASCAL datasets is for each CNN method that we now compare the *fc6* layer. We compare our approach with HOG-LDA [12], the triplet approach of [33], CliqueCNN [3], Imagenet pre-trained AlexNet [14], and RCNN [11]. In Tab. 5 we show the classification performance for all methods for $k = 5$ (for $k > 5$ there was only insignificant performance improvement). Our approach improves upon the initial similarities of the unsupervised triplet approach of [33] to yield a performance gain of 6% without requiring any supervision information or fine-tuning on PASCAL.

HOG-LDA	Wang et. al [33]	CliqueCNN[3]
0.1180	0.4501	0.4812
Wang et.al [33] + Ours	Alexnet [14]	RCNN [11]
0.5101	0.6160	0.6825

Table 5. Classification results for PASCAL VOC 2007

5. Conclusions

We have presented an unsupervised approach to similarity learning based on CNNs by framing it as a combination of surrogate classification tasks and poset ordering. This generalizes the widely used tuple and triplet losses to establish relations between large numbers of samples. Similarity learning then becomes a joint optimization problem of grouping samples into surrogate classes while learning the deep similarity encoding representation. In the experimental evaluation the proposed approach has shown competitive performance when compared to state-of-the-art results, learning fine-grained similarity relationships in the context of human pose estimation and object classification [3].

[3]This research has been funded in part by the Heidelberg Academy of Sciences. We are grateful to the NVIDIA corporation for donating a Titan X GPU.

References

[1] Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous systems, 2015. *Software available from tensorflow. org*, 1, 2015. 6

[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. June 2014. 7

[3] Miguel A Bautista, Artsiom Sanakoyeu, Ekaterina Sutter, and Björn Ommer. Cliquecnn: Deep unsupervised exemplar learning. *NIPS*, 2016. 2, 6, 7, 8

[4] Amir Beck and Luba Tetruashvili. On the convergence of block coordinate descent type methods. *SIAM journal on Optimization*, 23(4):2037–2060, 2013. 5

[5] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, pages 1422–1430, 2015. 1, 2, 4, 6

[6] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *NIPS*, pages 766–774, 2014. 2, 6

[7] A. Eigenstetter, M. Takami, and B. Ommer. Randomized max-margin compositions for visual recognition. In *CVPR '14*. 3

[8] I. El-Naqa, Y. Yang, N. P. Galatsanos, R. M. Nishikawa, and M. N. Wernick. A similarity learning approach to content-based image retrieval: application to digital mammography. *TMI*, 23(10):1233–1244, 2004. 1

[9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 8

[10] Pedro Felzenszwalb, David McAllester, and Deva Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, pages 1–8. IEEE, 2008. 5

[11] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014. 8

[12] Bharath Hariharan, Jitendra Malik, and Deva Ramanan. Discriminative decorrelation for clustering and classification. In *ECCV*, pages 459–472. Springer, 2012. 3, 6, 8

[13] Sam Johnson and Mark Everingham. Learning effective human pose estimation from inaccurate annotation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 6

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012. 5, 6, 7, 8

[15] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, volume 3, page 2, 2013. 2

[16] Tomasz Malisiewicz, Abhinav Gupta, and Alexei A Efros. Ensemble of exemplar-svms for object detection and beyond. In *ICCV*, pages 89–96. IEEE, 2011. 6

[17] Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2624–2637, 2013. 2

[18] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Interspeech*, volume 2, page 3, 2010. 5

[19] Martin R Min, Laurens Maaten, Zineng Yuan, Anthony J Bonner, and Zhaolei Zhang. Deep supervised t-distributed embedding. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 791–798, 2010. 2

[20] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *European Conference on Computer Vision*, pages 527–544. Springer, 2016. 1, 2, 4, 6, 7, 8

[21] Juan Carlos Niebles, Chih-Wei Chen, and Li Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, pages 392–405. Springer, 2010. 5

[22] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei Efros. Context encoders: Feature learning by inpainting. 2016. 1

[23] H. Pirsiavash and D. Ramanan. Parsing videos of actions with segmental grammars. In *CVPR*, 2014. 1

[24] V. Ramakrishna, D. Munoz, M. Hebert, J. A. Bagnell, and Y. Sheikh. Pose machines: Articulated pose estimation via inference machines. In *ECCV '14*. Springer, 2014. 7

[25] Oren Rippel, Manohar Paluri, Piotr Dollar, and Lubomir Bourdev. Metric learning with adaptive density discrimination. *arXiv preprint arXiv:1511.05939*, 2015. 2

[26] Sam Roweis, Geoffrey Hinton, and Ruslan Salakhutdinov. Neighbourhood component analysis. *Advances in Neural Information Processing Systems (NIPS)*, 17:513–520, 2005. 2

[27] J. Rubio, A. Eigenstetter, and B. Ommer. Generative regularization with latent topics for discriminative object recognition. *PR*, 48:3871–3880, 2015. 3

[28] Ruslan Salakhutdinov and Geoffrey E Hinton. Learning a nonlinear embedding by preserving class neighbourhood structure. In *AISTATS*, pages 412–419, 2007. 2

[29] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1653–1660, 2014. 7, 8

[30] Paul Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494, 2001. 5

[31] Pascal Vincent, Alexandre de Brébisson, and Xavier Bouthillier. Efficient exact gradient update for training deep networks with very large sparse targets. In *Advances in Neural Information Processing Systems*, pages 1108–1116, 2015. 4

[32] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *CVPR*, pages 1386–1393, 2014. 2

[33] X. Wang and A. Gupta. Unsupervised learning of visual representations using videos. In *ICCV*, 2015. 1, 2, 4, 8

[34] Yu-Xiong Wang and Martial Hebert. Learning from small sample sets by combining unsupervised meta-training with cnns. In *Advances in Neural Information Processing Systems*, pages 244–252, 2016. 2

[35] Jason Weston, Samy Bengio, and Nicolas Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *Proceedings of the International Joint Conference on Artificial Intelligence, IJCAI*, 2011. 2

[36] Hao Xia, Steven CH Hoi, Rong Jin, and Peilin Zhao. Online multiple kernel similarity learning for visual search. *TPAMI*, 36(3):536–549, 2014. 1

[37] Yangyang Xu and Wotao Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences*, 6(3):1758–1789, 2013. 5

[38] Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1, 4

[39] S. Zagoruyko and N. Komodakis. Learning to compare image patches via convolutional neural networks. In *CVPR '14*. 2