Multi-scale Matching Networks for Semantic Correspondence

Dongyang Zhao[1,2], Ziyang Song[1], Zhenghao Ji[1], Gangming Zhao[3], Weifeng Ge [*1,2], and Yizhou Yu[3]

[1]Nebula AI Group, School of Computer Science, Fudan University
[2]Shanghai Key Lab of Intelligent Information Processing
[3]Department of Computer Science, The University of Hong Kong

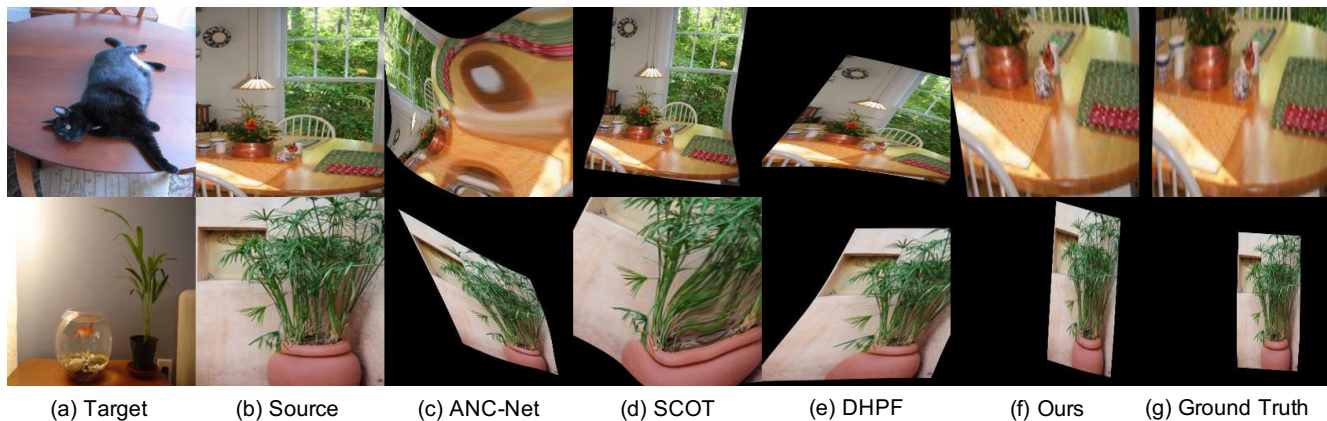| (a) Target | (b) Source | (c) ANC-Net | (d) SCOT | (e) DHPF | (f) Ours | (g) Ground Truth |

Figure 1. Dense correspondence prediction produced by state-of-the-art algorithms, including ANC-Net [26], SCOT [30], DHPF [35] and our multi-scale matching network. With the predicted key point pairs, images are warped with thin-plate splines algorithm [3].

Abstract

Deep features have been proven powerful in building accurate dense semantic correspondences in various previous works. However, the multi-scale and pyramidal hierarchy of convolutional neural networks has not been well studied to learn discriminative pixel-level features for semantic correspondence. In this paper, we propose a multi-scale matching network that is sensitive to tiny semantic differences between neighboring pixels. We follow the coarse-to-fine matching strategy and build a top-down feature and matching enhancement scheme that is coupled with the multi-scale hierarchy of deep convolutional neural networks. During feature enhancement, intra-scale enhancement fuses same-resolution feature maps from multiple layers together via local self-attention and cross-scale enhancement hallucinates higher-resolution feature maps along the top-down pathway. Besides, we learn complementary matching details at different scales thus the overall matching score is refined by features of different semantic levels gradually. Our multi-scale matching network can be trained end-to-end easily with few additional learnable parameters. Experimental results demonstrate that the proposed method achieves state-of-the-art performance on three popular benchmarks with high computational efficiency. The code has been released at https: //github.com/wintersun661/MMNet.

1. Introduction

Finding pixel-wise correspondences between a pair of semantically similar images has been a longstanding fundamental problem in computer vision. They have been proven useful for many tasks including optical flow [16, 45, 46], geometric matching [39, 32, 50], disparity estimation [36, 60], object recognition [9, 54, 58], semantic segmentation [17, 24] and etc. Due to large intra-class variations in color, scale, orientation, illumination and non-rigid deformations, the problem of semantic correspondence remains very challenging. With the breakthrough in representation learning, semantic correspondence has achieved impressive improvements in various scenarios.

Despite that deep features have improved matching accuracy significantly, the multi-scale and hierarchical structures of deep convolutional neural networks have not been explored thoroughly for semantic correspondence. It is

---
*Corresponding author: wfge@fudan.edu.cn

well-known that convolutional neural networks can extract features of different semantic levels in a bottom-up manner [57]. Bottom convolutional layers close to the input image act like low level feature descriptors, and are sensitive to colors, edges, textures and other low level statistics. Top convolutional layers contain high level semantics which are similar among neighboring points in feature maps. Methods like NC-Net [40], DualRC-Net [27] and GOCor [49] use the features from the topmost layer as the feature representation. However, in semantic correspondence, the ambiguity between neighboring pixels in the topmost layer leads to inferior performance. Hyperpixel flow [21] and its extension [35] combine features at different semantic levels to generate reliable feature representation and achieve improved results. However, they have not thoroughly exploited the multi-scale and hierarchical structure of deep convolutional neural networks.

Given an image pair, human usually tends to glance at the whole images, and then compare details carefully to establish semantic correspondence. It is similar to a coarse-to-fine matching scheme. In a convolutional neural network, neurons in the top layers have larger receptive fields while neurons in the bottom layers have relatively small receptive fields, which means top layers are rich in semantics but have relatively weak localization ability while the bottom layers are strong in localization but have less semantics. It will be helpful to follow the multi-scale and hierarchical structure of convolutional neural networks to find semantic correspondence in a coarse-to-fine manner.

In this paper, we propose a new multi-scale matching network to produce reliable semantic correspondence by integrating features of different semantic levels hierarchically and learn complementary matching details in a coarse-to-fine manner. The multi-scale matching network consists of an encoder and a decoder. The encoder is a typical convolutional neural network pretrained on the ImageNet ILSVRC dataset [43]. It contains many layers to capture semantic information at different levels. We divide the feature maps in the encoder into five convolutional groups with respect to their resolutions. The decoder has two top-down hierarchical enhancement pathways across different scales. The first one is the feature enhancement pathway which upsamples spatially coarser, but semantically stronger feature maps and fuse them with features from lateral connections to hallucinate higher resolution features. The second one is the matching enhancement pathway that learns finer and complementary matching details to enhance coarser matching results from a lower resolution. We start from the first layer in the decoder to generate the coarsest matching results, and upsample and enhance them with complementary matching details at different semantic levels.

To increase fine-grained details in feature maps, during intra-scale feature enhancement, we fuse all feature maps

from the same convolutional group in the encoder not just the feature map of the last layer. We also design a transformer with a local self-attention mechanism to enhance features that are discriminative among neighboring pixels. Besides, we supervise matching detail learning at different scales to make sure the network learns reliable semantic correspondences. Our multi-scale matching network adds relatively few learnable parameters with little extra computational cost, and can be trained in an end-to-end manner easily.

In summary, the main contributions of this work can be summarized as follows:

- We propose a multi-scale matching network that utilizes the multi-scale and hierarchical structure of deep convolutional neural network to learn semantic correspondences in a coarse-to-fine manner. Two top-down pathways in the decoder are built to couple the backbone encoder. The feature enhancement pathway increases the representation power of feature maps with intra-scale enhancement and cross-scale enhancement. The matching enhancement pathway learns matching details that are complementary to matching results from coarser levels.

- We design a novel intra-scale feature enhancement module that simultaneously fuses all the feature maps in each convolutional group and further increases the discriminative ability of the fused feature map with a local transformer.

- Experimental results demonstrate that our multi-scale matching network achieves state-of-the-art performance on multiple popular benchmarks, including PF-PASCAL [11], CUB [53] and SPair-71k [34].

2. Related Work

Semantic Correspondence. Methods for semantic correspondence can be roughly categorized into several groups: handcrafted feature based methods [31, 2, 42, 7, 48], learnable feature based methods [26, 27, 35, 49], graph matching and optimization based methods [52, 30, 58, 56], methods focusing on geometry displacement [5, 18, 12, 11, 50], and etc. Hand crafted features, such SIFT [31], HOG [47] and DAISY [48], design robust feature descriptors with low level statistics. In NC-Net [40], DualRC-Net [27] and GOCor [49], high level semantic features of convolutional neural networks are used to build dense correspondences beween image pairs. SCOT [30] and DeepEMD [58] formulate the semantic correspondence as an optimal transport problem and give closed-form solutions. PCA-GM [52] and other graph matching based methods focus on solving

(a) Multi-scale Matching Network

Image Pair   Convolutional Backbone   Feature Enhancement   Matching Enhancement   Semantic Correspondence

(b) Feature Enhancement

Inner-scale Enhancement

Residual Block 1 | SEM | Conv
Residual Block 2 | SEM | Conv
Residual Block 3 | SEM | Conv
⋮
Residual Block N | SEM | Conv

Local Self Attention

The L-th Convolution Group

Feature Maps at the Layer L-1
Dconv
Feature Concatenation
Conv
Feature Maps at the Layer L

(c) Matching Enhancement

4-D Matching Scores at the Layer L-1
Bicubic Interpolation

Source Feature Volume
4-D Complementary Matching Scores
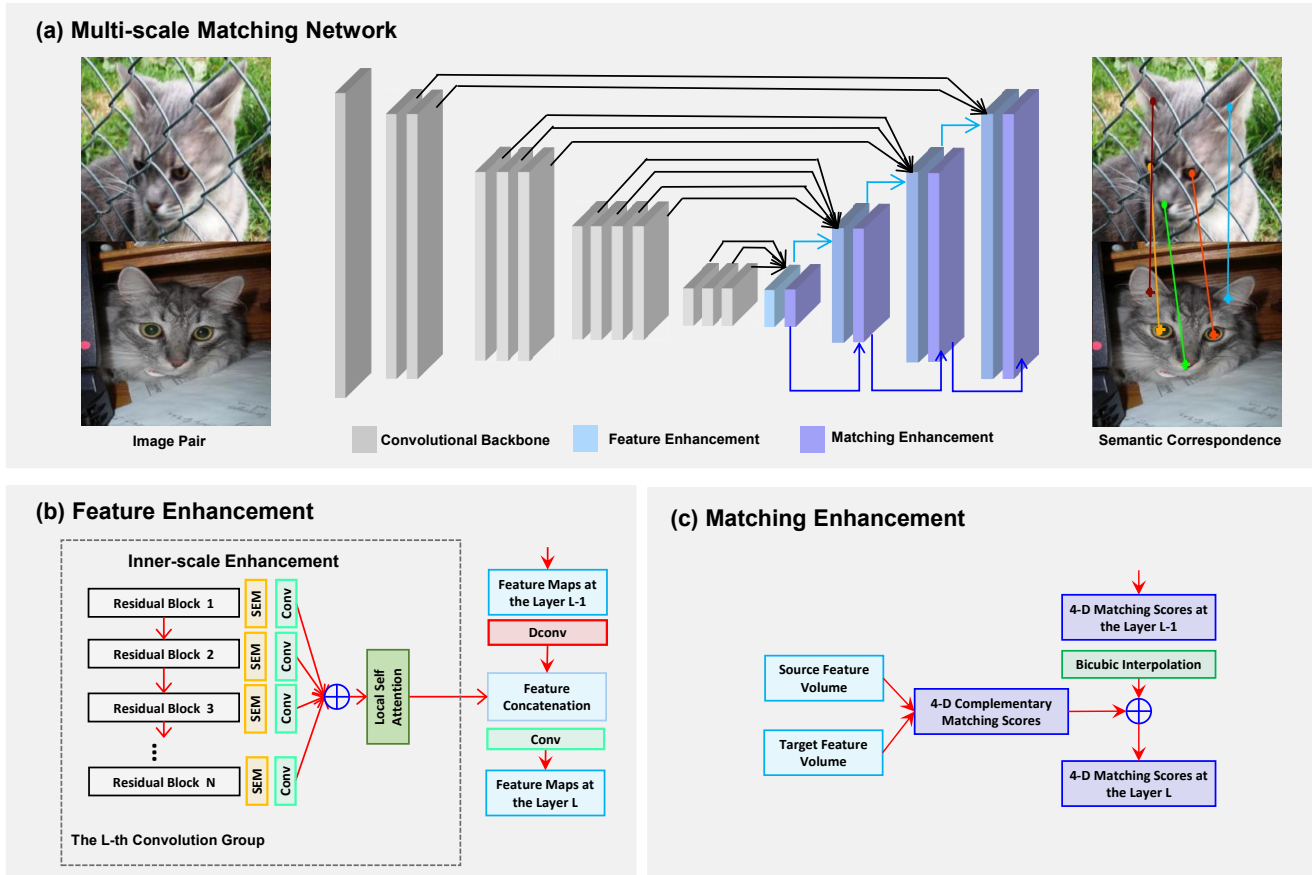Target Feature Volume

4-D Matching Scores at the Layer L

Figure 2. Illustration of the multi-scale matching network. Our multi-scale matching network contains a convolutional backbone, and a top-down feature and matching enhancement pathway. In feature enhancement, coarser but semantically richer features are upscaled and combined with finer but semantically weaker features to enhance the discrminaitve ability of pixel-level features. Then the matching enhancement module directly upsamples the matching results of the previous scale and adds it with the current matching details to learn complementary correspondences across scales.

a general quadratic assignment programming (QAP) problem to get matching results. Besides, PHM [5, 11] and SCNet [12] develop the probabilistic Hough matching in a Bayesian probability framework to model the geometry displacement of objects between two images.

In this paper, we learn semantic correspondence in a coarse-to-fine manner with a top-down matching enhancement scheme. By such hierarchical matching scheme, semantics at different levels and scales are fused and enhanced to get accurate pixel-wise correspondences.

Multi-scale Feature Fusion. Multi-scale feature fusion can improve the representation ability of features in many tasks, including object detection [28, 61], semantic segmentation [41, 29] and semantic correspondence [33, 35]. Feature pyramid networks (FPN [28]) build a decoder with a top-down pathway and lateral connections, and achieve impressive results on object detection. Hyperpxiel flow (HPF [33]) searches the most informative convolutional feature to get superior results on semantic correspondence. Its

extension Hypercolumns [35] designs a learning algorithm to select convolutional features in different layers in a much more efficient learning scheme.

Different from FPN [28], in the top-down pathway of our multi-scale matching network, feature maps in every layer of a convolutional group are fused to generate the lateral connection, not just the output of the last layer. Compared with HPF [33] and Hypercolumns [35], our multi-scale learning scheme is much more flexible with simple top-down and lateral connections, and can benefit from the multi-scale and pyramid hierarchy more efficiently..

4-D Correlation. The 4-D correlation between two feature volumes is popular in semantic correspondence learning which calculates the matching scores densely. NC-Net [40] analyzes neighborhood consensus patterns in the 4-D correlation space to get reliable dense correspondences. ANC-Net [26] introduces a set of non-isotropic 4-D convolution layers to capture adaptive neighborhood consensus. In this paper, we don't normalize the feature maps with L_2 nor-

malization as that in NC-Net [40] and ANC-Net [26]. We simply use the 4-D correlation tensor of two feature maps which stores the pair-wise scalar products as the matching score tensor. Then this 4-D correlation tensor is normalized with softmax to get the matching probability of every feature point. Experiments demonstrate that our proposed method without L_2 normalization can achieve impressive results.

Transformer. Transformers have led to a series of breakthroughs in computer vision [62, 4, 59] and natural language processing [51, 8, 1]. In [51], elements in a sequence are encoded with a self-attention mechanism uniformly. While, in our local self attention, only neighborhood pixels are considered to enhancement the local patterns. Experimental results show that this local property in our local self attention works quite well for semantic correspondence.

3. Multi-scale Matching Networks

Multi-scale matching networks utilize the multi-scale and hierarchical structures of deep convolutional neural networks to get discriminative pixel-level semantics for semantic correspondence. Figure 2 gives an overview of the proposed method. Given a pair of images $\left(\boldsymbol{I}^s, \boldsymbol{I}^t\right)$ and the ground truth of their matched key points $\mathcal{M}_{gt} = \left\{m_i = \left(\boldsymbol{p}_i^s, \boldsymbol{p}_i^t\right) | i = 1, ..., K\right\}$, our multi-scale matching network adopts a top-down feature enhancement scheme and a coarse-to-fine matching enhancement scheme respectively. In the feature enhancement, we design a local self attention which models dependencies between neighborhood neurons to reduce the semantic ambiguity. In the matching enhancement, we learn the matching details that are complementary at different semantic levels. Besides, different from that in [40, 26], feature volumes of two images are multiplied together directly without L_2 normalization to get 4-D matching scores. This scheme ensures the pixel-level similarity can be learnt with deep neural networks efficiently. Then we can train the multi-scale matching network by adding supervision at different scales in an end-to-end manner.

Network Structure. A typical convolutional neural network for image classification [23, 14] usually has five convolutional groups with different resolutions. Here we adopt ResNet [14] as our backbone and follow the feature pyramid network (FPN [28]) to broadcast semantics from high level layers to low level layers with a top-down pathway and lateral connections as shown in Figure 2 (a). Then in the decoding part, we get 4 feature maps with successive increasing resolutions by a scaling factor 2.

Feature Enhancement. There are feature enhancements both in the same scale and across different scales. During intra-scale enhancement, unlike FPN [28] which only takes the feature map of the last residual block in each convolutional group, we simultaneously fuse the feature maps of all residual blocks in the same group to capture semantics at different levels. To enlarge the receptive field of a single neuron and capture semantics at different scales, every feature map passes a scale enhancement module (SEM [13]) before fusion. Then these feature maps are simply added together and go into a local self attention module to finish intra-scale enhancement.

The local self attention module is designed to specialize every feature point. As common transformers [51, 59], our local self attention operates on *queries*(Q), *keys*(K) and *values*(V) with an input feature map $\boldsymbol{X} \in \mathbb{R}^{C \times H \times W}$, and output a transformed version $\widetilde{\boldsymbol{X}}$ with the same shape as $\boldsymbol{X}$. For every location in $\boldsymbol{X}$, we select its $r \times r$ neighborhood to conduct the self-attention. Feature vectors of the neighborhood locations in $\boldsymbol{X}$ are collected, and then we get a neighborhood feature tensor $\boldsymbol{X}' \in \mathbb{R}^{C \times H \times W \times r \times r}$. We send the input feature map $\boldsymbol{X}$ into the *query* transformation function $\mathcal{F}_q$, and send the neighborhood feature tensor $\boldsymbol{X}'$ into the *key* and *value* transformation functions $\mathcal{F}_k/\mathcal{F}_v$ respectively. The *query*/*key*/*value* transformation functions are implemented with 1×1 convolutions followed by ReLU activations respectively. The local interaction of our transformer is written as,

$$\widetilde{\boldsymbol{X}}_i = \boldsymbol{X}_i + \mathcal{G}\left(\mathcal{F}_v\left(\boldsymbol{X}_i'\right) \delta \left(\mathcal{F}_q\left(\boldsymbol{X}_i\right)^{\mathrm{T}} \mathcal{F}_k\left(\boldsymbol{X}_i'\right)\right)^{\mathrm{T}}\right),$$
$$(1)$$

where $\boldsymbol{X}_i$ is the feature vector of the ith grid cell in $\boldsymbol{X}$, $\boldsymbol{X}_i'$ is the neighborhood feature tensor of the ith grid cell, $\mathcal{F}_q\left(\boldsymbol{X}_i\right) \in \mathbb{R}^{C'}$ is i^{th} *query*, $\mathcal{F}_k\left(\boldsymbol{X}_i'\right) \in \mathbb{R}^{C' \times r^2}$ and $\mathcal{F}_v\left(\boldsymbol{X}_i'\right) \in \mathbb{R}^{C' \times r^2}$ is i^{th} *key*/*value* pair, δ is the SoftMax operation, and $\mathcal{G}$ is a transformation function implemented with 1×1 convolution.

After local self feature enhancement, we perform cross scale feature enhancement. Like the top-down pathway in FPN [28], the feature map from the previous matching stage are upsampled with a deconvolutional layer and are concatenated with $\widetilde{\boldsymbol{X}}$ before going through another convolutional layer. Then the enhanced feature map can be used to calculate the matching scores of an image pair and enhance features in the next stage.

Matching Enhancement. Unlike many cascaded methods such as FPN [28] and BDCN [13] where results of different scales are jointly fused, we enhance the matching result in a top-down manner by learning matching complement at different scales. For a pair of images $\left(\boldsymbol{I}^s, \boldsymbol{I}^t\right)$, denote their feature maps at the lth scale with $\boldsymbol{X}_l^s$ and $\boldsymbol{X}_l^t$ respectively. $\boldsymbol{X}_l^s$ and has the resolution $H_l^s \times W_l^s$, and $\boldsymbol{X}_l^t$ has the resolution $H_l^t \times W_l^t$. We calculate the exhaustive pair-

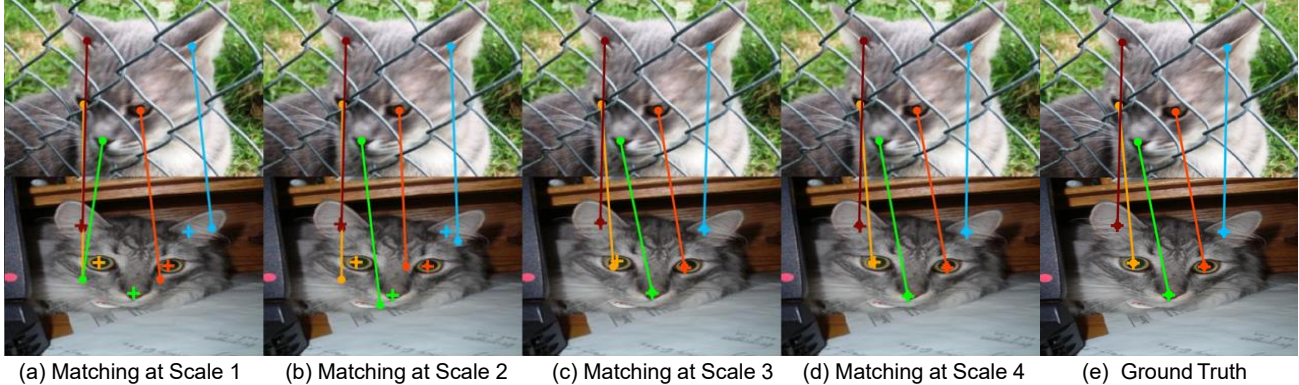| (a) Matching at Scale 1 | (b) Matching at Scale 2 | (c) Matching at Scale 3 | (d) Matching at Scale 4 | (e) Ground Truth |

Figure 3. Matching results at different feature resolutions. From left to right, displacements between predictions and the destination points are reduced with the increase of the feature resolution.

wise products between $\boldsymbol{X}_l^s$ and $\boldsymbol{X}_l^t$, and store the results in a 4-D tensor $\widetilde{\boldsymbol{S}}_l \in \mathbb{R}^{H_l^s \times W_l^s \times H_l^t \times W_l^t}$ referred to as the matching score. $\boldsymbol{S}_l(i,j,m,n)$ is the match score between the (i,j) grid cell in $\boldsymbol{X}_l^s$ and the (m,n) grid cell in $\boldsymbol{X}_l^t$. Given the matching score $\boldsymbol{S}_{l+1} \in \mathbb{R}^{H_{l+1}^s \times W_{l+1}^s \times H_{l+1}^t \times W_{l+1}^t}$ from the previous scale, the matching complementation is conducted as follows,

$$\boldsymbol{S}_l = \widetilde{\boldsymbol{S}}_l + \mathcal{U}\left(\boldsymbol{S}_{l+1}\right), \tag{2}$$

where $\mathcal{U}$ is the 4-D bicubic upscaling interpolation. Then the l-th scale just learns the matching residuals that are complementary with the matching results in the $(l+1)$-th scale. Note that we start from the 5-th scale with the highest semantic level where $\boldsymbol{S}_5 = \widetilde{\boldsymbol{S}}_5$, and end at the 2-nd scale. Figure 3 visualizes the improvements caused by the matching enhancement. Given these four matching results, we test their performance on the validation set, and select the scale that has the best performance to conduct testing.

Correspondence Learning with Rich Supervision. Labeling dense semantic correspondences of image pairs requires huge amount of human labours which is impractical in real applications. We evaluate the effectiveness of the proposed method on existing datasets with sparse keypoint annotations including PF-PASCAL [11], CUB [53] and SPair-71k [34]. The sparse key-point annotation stores many one-tn-one mapping between images, where each mapping can be viewed as a probability distribution of a pixel in a source image $\boldsymbol{I}^s$ matched with all pixels in a target image $\boldsymbol{I}^t$. Then these annotations can be utilized in a straightforward way to train a CNN model for semantic matching by minimizing the distance between the matching distribution and the ground-truth distribution. To benefit from the multi-scale matching mechanism further, we design loss functions at every scale to supervise the learning process. This rich supervision style leads to much more accurate matching results as stated in Table 3.

For a key point $\boldsymbol{p}_i^s$ in the source image, the matching score $\boldsymbol{S}_l(\boldsymbol{p}_i^s) \in \mathbb{R}^{H_l^t \times W_l^t}$ with the target image in the l-th scale is denoted in a 2-D form. Since deep features have very strong discriminative ability, we simply get the matching probability matrix $\boldsymbol{P}_l(\boldsymbol{p}_i^s)$ by applying the SoftMax operation spatially. We first rescale the key-points in $\mathcal{M}_{gt}$ to the same resolution as the feature maps at different scales. Then following ANC-Net [26], we pick its four nearest neighbours and set their probability according to distance to establish the 2-D ground-truth matching probabilities at every scale. Then we apply 2-D Gaussian smoothing of size 3 on that probability map. Our training objectives for semantic matching is then,

$$\mathcal{L} = \sum_l \alpha_l \left[\mathcal{B}\left(\boldsymbol{P}_l(\boldsymbol{p}_i^s), \widetilde{\boldsymbol{P}}_l(\boldsymbol{p}_i^s)\right) + \mathcal{B}\left(\boldsymbol{P}_l(\boldsymbol{p}_i^t), \widetilde{\boldsymbol{P}}_l(\boldsymbol{p}_i^t)\right) \right], \tag{3}$$

where α_l (=1) is the weight at the l scale, $\mathcal{B}$ is the binary cross entropy loss, and $\widetilde{\boldsymbol{P}}_l(\boldsymbol{p}_i^s)$ and $\widetilde{\boldsymbol{P}}_l(\boldsymbol{p}_i^t)$ are the ground-truth probability map of the key-point pair $(\boldsymbol{p}_i^s, \boldsymbol{p}_i^t)$.

4. Experiments

Dataset. We conduct experiments on three popular benchmarks for semantic correspondence: PF-PASCAL [11], CUB [53] and SPair-71k [34]. The PF-PASCAL contains 1351 image pairs which are selected from all the 20 categories in PASCAL VOC [10]. We split the dataset as done in [12] where approximately 700 image pairs are used for training, 300 image pairs are used for validation and 300 image pairs are used for test. The CUB dataset [53] contains 11,788 images of 200 bird species with large intra-class variations. Each image is annotated with the locations of 15 key-parts. Following the protocol in [26], we randomly sample 10,000 pairs from the CUB as training data and use the same test set provided by [22]. SPair-71k is composed of total 70,958 image pairs in 18 categories with large view-point and scale variations. We

Methods	PF-PASCAL			CUB	time
	0.05	0.1	0.15	0.1	(ms)
PF$_{HOG}$ [11]	31.4	62.5	79.5	-	-
CNNGeo$_{ResNet-101}$ [38]	41.0	69.5	80.4	-	-
A2Net$_{ResNet-101}$ [44]	42.8	70.8	83.3	-	-
SFNet$_{ResNet-101}$ [25]	53.6	81.9	90.6	-	-
DCTM$_{CAT-FCSS}$ [20]	34.2	69.6	80.2	-	-
WeakAlign$_{ResNet-101}$ [39]	49.0	74.8	84.0	-	-
SCNet$_{VGG-16}$ [12]	36.2	72.2	82.0	-	-
RTNs$_{ResNet-101}$ [19]	55.2	75.9	85.2	-	-
UCN$_{GoogLeNet}$ [6]	-	55.6	-	48.3	-
UCN$_{ResNet-101}$ [6]	-	75.1	-	52.1	-
NC-Net$_{ResNet-101}$ [40]	54.3	78.9	86.0	64.7	393
DCCNet$_{ResNet-101}$ [15]	55.6	82.3	90.5	66.1	-
HPF$_{ResNet-50}$ [33]	60.5	83.4	92.1	-	-
HPF$_{ResNet-101}$ [33]	60.1	84.8	92.7	-	-
HPF$_{ResNet-101-FCN}$ [33]	63.5	88.3	95.4	-	-
DHPF$_{ResNet-50}$ [35]	72.6	**88.9**	**94.3**	-	55
DHPF$_{ResNet-101}$ [35]	75.7	**90.7**	**95.0**	-	95
SCOT$_{ResNet-101}$ [30]	63.1	85.4	92.7	-	180
SCOT$_{ResNet-101-FCN}$ [30]	67.3	88.8	95.4	-	109
ANC-Net$_{ResNet-101}$ [26]	-	83.7	-	69.6	600
ANC-Net$_{ResNeXt-101}$ [26]	-	88.7	-	74.1	-
ANC-Net$_{ResNet-101-FCN}$ [26]	-	86.1	-	72.4	-
MMNet$_{ResNet-50}$	**75.3**	88.0	93.2	**80.6**	51
MMNet$_{ResNet-101}$	**77.6**	89.1	94.3	**81.8**	86
MMNet$_{ResNeXt-101}$	**78.9**	90.3	94.4	**83.1**	101
MMNet$_{ResNet-101-FCN}$	**81.1**	**91.6**	**95.9**	**87.0**	87

Table 1. Comparison with state-of-the-art algorithms in PCK and speed on PF-PASCAL [11] and CUB [53] dataset. Subscripts of the method names indicates the backbone used.

use the same split proposed in [34] where 53340, 5384, 12234 image pairs are used for training, validation and testing respectively.

Evaluation metric. Performances of different methods are evaluated using the percentage of correct key-points (PCK@α). A point is considered *correct* if the predicted point is within the circle of radius $\alpha \times d$ centering at the ground-truth point, where d is the longer side of an image or an object bounding box as in [12, 40, 26, 30, 35].

Implementation Details For fair comparison with state-of-the art methods, we use four different backbones including ResNet-50 [14], ResNet-101 [14], ResNeXt-101 [55] and ResNet101-FCN [14]. All backbone networks are pre-trained on Image-Net1k classification set [23] and then fine-tuned for corrrespondence task.

The multi-scale matching network structure is visualized in Figure 2. We only introduce additional parameters in feature enhancement. As shown in Figure 2 (b), we have many SEMs each of which is followed by a 1×1 convolutional layer. We upscale the low resolution feature with a 4×4 deconvolutional layer whose stride is 2 at different scales. Then the upsampled feature map is concatenated with the output of the intra-scale feature enhancement, and pass a
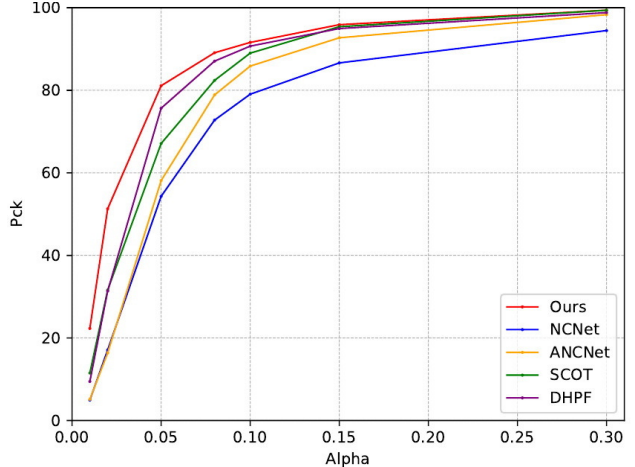

Figure 4. The PCK-α curves of our method and compared works on PF-PASCAL [11].

3×3 convolution layer. Note that the SEM in our MM-Net is in the same settings as in BDCN [13], and the output channel numbers of both the convolutional layers and the deconvolutional layers is set to 21 to save computation cost.

During the training, we adopt SGD with momentum as our optimizer. The learning rate is set to 0.0005 for initialization and is decreased by 10 times every 10000 iterations. Momentum and weight decay are set to 0.9 and 0.0002 respectively. Learning rate is decreased by 10 times every 10,000 iterations. The batch-size is set to 5 for all experiments. The training will converge within 10000, 32000 and 30000 iterations for PF-PASCAL [11], CUB [53] and SPair-71k [34] respectively. All experiments are implemented with PyTorch [37], and run on NVidia TITAN RTX GPUs.

4.1. Comparisons with State-of-the-art Methods

For PF-PASCAL[11], our MMNet with ResNet101-FCN as the backbone outperforms all previous state-of-art methods with 81.1% PCK@0.05, 91.6% PCK@0.1, and 95.9% PCK@0.15. When compared with ANC-Net [26] which also conducts end-to-end training with 4-D correlation, MMNet gets 5.4%, 1.6%, and 5.5% increases on PCK@0.1 with three different backbones respectively. It demonstrates the effectiveness of the multi-scale feature learning and matching complementation. When compared with previous best SCOT [30] with ResNet101-FCN as the backbone, we achieve a significant improvement on PCK@0.05 by 13.8%. We attribute this to that the end-to-end training of deep neural networks has a higher efficiency that optimization based methods to enforce one-on-one matching with discriminative features. We also compare with the multi-scale feature fusion based methods HPF [33] and DHPF [35]. MMNet with ResNet101-FCN as the backbone outperforms HPF with the same backbone by 17.6% PCK@0.05, 3.3%

Methods	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	dog	horse	mbike	person	plant	sheep	train	tv	all
CNNGeo [38]	23.4	16.7	40.2	14.3	36.4	27.7	26.0	32.7	12.7	27.4	22.8	13.7	20.9	21.0	17.5	10.2	30.8	34.1	20.6
A2Net [44]	22.6	18.5	42.0	16.4	37.9	30.8	26.5	35.6	13.3	29.6	24.3	16.0	21.6	22.8	20.5	13.5	31.4	36.5	22.3
WeakAlign [39]	22.2	17.6	41.9	15.1	38.1	27.4	27.2	31.8	12.8	26.8	22.6	14.2	20.0	22.2	17.9	10.4	32.2	35.1	20.9
NC-Net [40]	17.9	12.2	32.1	11.7	29.0	19.9	16.1	39.2	9.9	23.9	18.8	15.7	17.4	15.0	14.8	9.6	24.2	31.1	20.1
HPF [33]	25.3	18.5	47.6	14.6	37.0	22.9	18.3	51.1	16.7	31.5	30.8	19.1	23.7	23.8	23.5	14.4	30.8	37.2	27.2
HPF [33]	25.2	18.9	52.1	15.7	38.0	22.8	19.1	52.9	17.9	33.0	32.8	20.6	24.4	27.9	21.1	1.9	31.5	35.6	28.2
DHPF [35]	38.4	23.8	**68.3**	18.9	42.6	27.9	20.1	61.6	22.0	46.9	46.1	33.5	27.6	**40.1**	27.6	28.1	49.5	46.5	37.3
SCOT [30]	34.9	20.7	63.8	21.1	43.5	27.3	21.3	**63.1**	20.0	42.9	42.5	31.1	29.8	35.0	27.7	24.4	48.4	40.8	35.6
MMNet	43.5	27.0	62.4	27.3	40.1	50.1	37.5	60.0	21.0	56.3	50.3	41.3	30.9	19.2	30.1	33.2	64.2	43.6	40.9
MMNet-FCN	**55.9**	**37.0**	65.0	**35.4**	**50.0**	**63.9**	**45.7**	62.8	**28.7**	**65.0**	**54.7**	**51.6**	**38.5**	34.6	**41.7**	**36.3**	**77.7**	**62.5**	**50.4**

Table 2. Comparisons on SPair-71k [34] with state-of-art methods. The backbone in methods listed is ResNet101 [14]. The best results are reported in bold.

PCK@0.1 and 0.5% PCK@0.15. Since HPF doesn't conduct end-to-end training, it is reasonable that our MMNet gets better results. DHPF [35] selects features from the backbone, and get slightly better results on PCK@0.1 and PCK@0.15 with ResNet50, ResNet101 backbone. However, when $\alpha = 0.05$, our MMNet get 1.9% improvement on PCK. It may because MMNet are much more sensitive to small difference between neighborhood, and thus get better matching results with much more strict matching criteria. For CUB[53], our MMNet outperforms all state-of-art algorithms with 87.0% on PCK@0.1 and achieve prominent betters on three different backbone compared with ANC-Net [26].For SPair-71k[34], our MMNet with ResNet101-FCN backbone outperforms the state-of-art algorithms by at least 13.1% on PCK@0.1, which is a huge improvement. Among all listed methods in Table 1, our algorithm surpasses all state-of-art by a large margin on 15 of the 18 classes. This proves the effectiveness and robustness of our MMNet in establishing reliable matching.

Figure 4 shows the results in comparison with state-of-art method with varying α. When α is small, only points matched with the destination point closely is treated as a correct match, otherwise failure. When we increase α, lager matching displacement will be allowed. It can be found that our MMNet achieves the best performance when α varies from 0.02 to 0.3. When α varies from 0.02 to 0.1, all algorithms will get improvements on PCK fast. It means the allowed match displacement influences the performance greatly. When α varies from 0.15 to 0.3, all methods get almost the same results with very high matching accuracy. This indicates too large α can not be used to measure the performance of different methods accurately. When α varies from 0.01 to 0.05, our MMNet outperforms other state-of-art methods by a clear margin all the time. It indicates the strong ability of MMNet in identifying neighborhood pixels.

Besides, we also report the running speed to compare the computational efficiency of state-of-art methods. In Table 1, we get the comparable test speed with the previous best DHPF [35], and are much faster than other methods.

Methods	0.05	0.1	0.15
MMNet	81.1	91.6	95.9
MMNet w/o local self attention	79.9	90.7	95.1
MMNet w/o dense connections	74.9	88.6	93.9
MMNet w/o cross-scale fusion	78.5	89.7	94.3
MMNet w/o complementation	80.0	89.3	94.3
MMNet w/o rich supervision	31.4	45.6	57.0
MMNet w/o last layer	77.9	90.9	94.6

Table 3. Ablation results on various setting on PF-PASCAL [11] with ResNet101-FCN as backbone.

4.2. Module Analysis with Ablations

To investigate the effectiveness of different modules, we conduct ablation study by replacing or removing a single component. All experiments are conducted on PF-PASCAL [11] with ResNet101-FCN as the backbone. PCKs are evaluated with $\alpha = 0.05$, 0.1 and 0.15. Results are listed in Table 3. First, we remove the local self attention, the performance drops by 1.2% on PCK@0.05 which indicates that the contextual information around neighborhood pixels is very important. Then we remove the dense connections and only get the output of the last layer in every convolutional group, the performance of PCK@0.05 is 74.9% which is 6.2% lower than MMNet with dense connections. It shows fusing information at different layers in a convolutional group is greatly helpful. When we remove the cross-scale feature fusion, PCK@0.05, PCK@0.10 and PCK@0.15 decrease by 2.6%, 1.9% and 1.6% respectively. If we don't conduct the complementary matching learning and get the matching score at every layer independently, PCK@0.05 drops by 1%. At last, if we only supervise the learning at the last matching complementation layer with the highest resolution, the performance drops to 31.4% on PCK@0.05 which is 49.7% lower than the MMNet with rich supervision during training. It may indicate that supervision in multiple scales are very important to learn semantics at different levels.

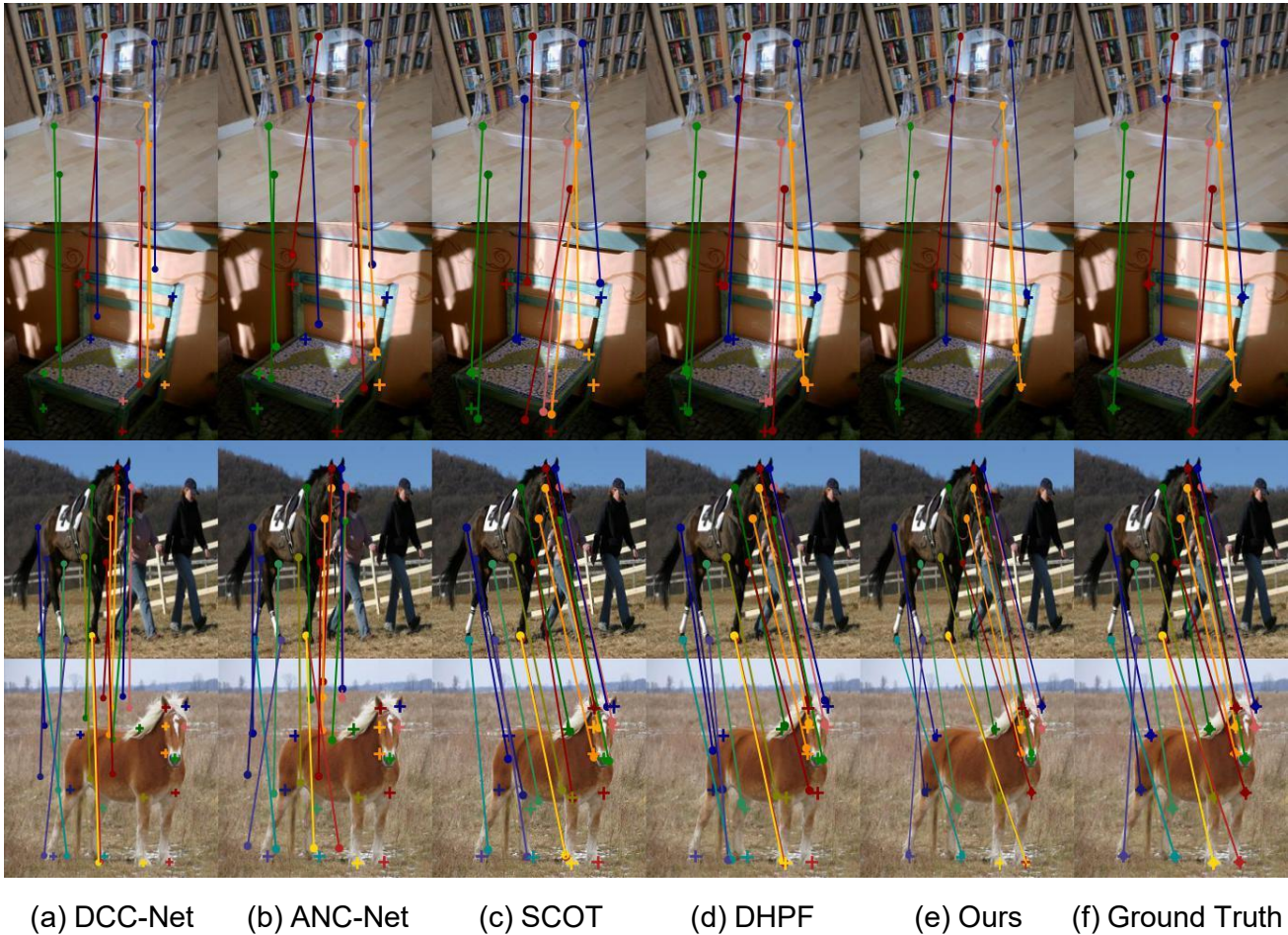(a) DCC-Net   (b) ANC-Net   (c) SCOT   (d) DHPF   (e) Ours   (f) Ground Truth

Figure 5. Visualization of the semantic correspondence. The odd rows are the source image, and the even rows are the target images. Destination key point are denoted with crosses. From left to right: (a) DCC-Net [15], (b) ANC-Net [26], (c) SCOT [30], (d) DHPF [35], (e) ours MMNet and (f) the ground truth.

4.3. Qualitative Results and Visualization

We visualize the correspondence result by drawing the point-to-point matches and warping images with the predicted key point pairs respectively. In Figure 5, the point-to-point matches are drawn by linking key point pairs with line segments. The ground truth matching is given at first as reference for visual comparison. It can be find that our MM-Net matches all key points on horses correctly. Other state-of-the art methods, such as DCC-Net [15], ANC-Net [26], SCOT [30] and DHPF [35], will lead to large mismatch displacement or many-to-one match. In Figure 1, images are warped based on matched key point pairs. For convenience, we warp the ground truth annotations as the reference. It can be found that our MMNet can matches the objects in images accurately. Especially in the first row, tables in the source and target images has very large viewpoint and appearance variations. But our MMNet can still match the corresponding key points accurately when other methods fail.

5. Conclusion

In this paper, we propose a multi-scale matching network that are coupled with the multi-scale and pyramidal hierarchy of deep convolutional neural networks to match semantic meaningful points in a coarse-to-fine manner. Our multi-scale matching network learn discriminative pixel-level semantics by a top-down feature and matching enhancement scheme. Thus discriminative features are identified and fused during the complementary matching learning process. To strengthen the representatve ability of individual pixels, we introduce a local self attention module to encode local contextual information to disambiguate the feature representation of neighborhood pixels. Extensive experiments on several popular benchmarks demonstrate that the proposed MMNet outperform existing state-of-the-art semantic correspondence algorithms. However, how to get reliable correspondence that can handle drastically changes in real applications still remains to be an open problem.

References

[1] Christoph Alt, Marc Hübner, and Leonhard Hennig. Fine-tuning pre-trained transformer language models to distantly supervised relation extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1388–1398, Florence, Italy, July 2019. Association for Computational Linguistics. 4

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006. 2

[3] Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on pattern analysis and machine intelligence*, 11(6):567–585, 1989. 1

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 4

[5] Minsu Cho, Suha Kwak, Cordelia Schmid, and Jean Ponce. Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2, 3

[6] Christopher B Choy, JunYoung Gwak, Silvio Savarese, and Manmohan Chandraker. Universal correspondence network. In *Advances in Neural Information Processing Systems 30*. 2016. 6

[7] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee, 2005. 2

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 4

[9] Olivier Duchenne, Armand Joulin, and Jean Ponce. A graph-matching kernel for object categorization. In *2011 International Conference on Computer Vision*, pages 1792–1799. IEEE, 2011. 1

[10] Mark Everingham, S. M. Eslami, Luc Gool, Christopher K. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vision*, 111(1):98–136, Jan. 2015. 5

[11] Bumsub Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow: Semantic correspondences from object proposals. *IEEE transactions on pattern analysis and machine intelligence*, 40(7):1711–1725, 2017. 2, 3, 5, 6, 7

[12] Kai Han, Rafael S Rezende, Bumsub Ham, Kwan-Yee K Wong, Minsu Cho, Cordelia Schmid, and Jean Ponce. Scnet: Learning semantic correspondence. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1831–1840, 2017. 2, 3, 5, 6

[13] Jianzhong He, Shiliang Zhang, Ming Yang, Yanhu Shan, and Tiejun Huang. Bi-directional cascade network for perceptual edge detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3828–3837, 2019. 4, 6

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4, 6, 7

[15] Shuaiyi Huang, Qiuyue Wang, Songyang Zhang, Shipeng Yan, and Xuming He. Dynamic context correspondence network for semantic alignment. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2010–2019, 2019. 6, 8

[16] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. Lite-flownet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8981–8989, 2018. 1

[17] Junhwa Hur and Stefan Roth. Joint optical flow and temporally consistent semantic segmentation. In *European Conference on Computer Vision*, pages 163–177. Springer, 2016. 1

[18] Angjoo Kanazawa, David W Jacobs, and Manmohan Chandraker. Warpnet: Weakly supervised matching for single-view reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3253–3261, 2016. 2

[19] Seungryong Kim, Steve Lin, Sang Ryul Jeon, Dongbo Min, and Kwanghoon Sohn. Recurrent transformer networks for semantic correspondence. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6126–6136, December 2018. 6

[20] Seungryong Kim, Dongbo Min, Stephen Lin, and Kwanghoon Sohn. Dctm: Discrete-continuous transformation matching for semantic flow. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4539–4548, 2017. 6

[21] Tao Kong, Anbang Yao, Yurong Chen, and Fuchun Sun. Hypernet: Towards accurate region proposal generation and joint object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 845–853, 2016. 2

[22] Jonathan Krause, Hailin Jin, Jianchao Yang, and Fei-Fei Li. Fine-grained recognition without part annotations. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5546–5555, 2015. 5

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 4, 6

[24] Mans Larsson, Erik Stenborg, Lars Hammarstrand, Marc Pollefeys, Torsten Sattler, and Fredrik Kahl. A cross-season correspondence dataset for robust semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9532–9542, 2019. 1

[25] Junghyup Lee, Dohyung Kim, Jean Ponce, and Bumsub Ham. Sfnet: Learning object-aware semantic correspondence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2278–2287, 2019. 6

[26] Shuda Li, Kai Han, Theo W Costain, Henry Howard-Jenkins, and Victor Prisacariu. Correspondence networks

with adaptive neighbourhood consensus. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10196–10205, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[27] Xinghui Li, Kai Han, Shuda Li, and Victor Prisacariu. Dual-resolution correspondence networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 2

[28] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 3, 4

[29] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8759–8768, 2018. 3

[30] Yanbin Liu, Linchao Zhu, Makoto Yamada, and Yi Yang. Semantic correspondence as an optimal transport problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4463–4472, 2020. 1, 2, 6, 7, 8

[31] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 2

[32] Iaroslav Melekhov, Aleksei Tiulpin, Torsten Sattler, Marc Pollefeys, Esa Rahtu, and Juho Kannala. Dgc-net: Dense geometric correspondence network. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1034–1042. IEEE, 2019. 1

[33] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Hyperpixel flow: Semantic correspondence with multi-layer neural features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3395–3404, 2019. 3, 6, 7

[34] Juhong Min, Jongmin Lee, J. Ponce, and Minsu Cho. Spair-71k: A large-scale benchmark for semantic correspondence. *ArXiv*, abs/1908.10543, 2019. 2, 5, 6, 7

[35] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Learning to compose hypercolumns for visual correspondence. In *ECCV 2020-16th European Conference on Computer Vision*. Springer, 2020. 1, 2, 3, 6, 7, 8

[36] Jiahao Pang, Wenxiu Sun, Jimmy SJ Ren, Chengxi Yang, and Qiong Yan. Cascade residual learning: A two-stage convolutional neural network for stereo matching. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 887–895, 2017. 1

[37] Adam Paszke, S. Gross, Soumith Chintala, G. Chanan, Edward Yang, Zachary Devito, Zeming Lin, Alban Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017. 6

[38] Ignacio Rocco, Relja Arandjelović, and Josef Sivic. Convolutional neural network architecture for geometric matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (41):2553–2567, 2018. 6, 7

[39] Ignacio Rocco, Relja Arandjelović, and Josef Sivic. End-to-end weakly-supervised semantic alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6917–6925, 2018. 1, 6, 7

[40] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. In *Proc. NIPS*, 2018. 2, 3, 4, 6, 7

[41] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 3

[42] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee, 2011. 2

[43] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 2

[44] Paul Hongsuck Seo, Jongmin Lee, Deunsol Jung, Bohyung Han, and Minsu Cho. Attentive semantic alignment with offset-aware correlation kernels. In *ECCV*, 2018. 6, 7

[45] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018. 1

[46] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Models matter, so does training: An empirical study of cnns for optical flow estimation. *IEEE transactions on pattern analysis and machine intelligence*, 42(6):1408–1423, 2019. 1

[47] Tatsunori Taniai, Sudipta N Sinha, and Yoichi Sato. Joint recovery of dense correspondence and cosegmentation in two images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4246–4255, 2016. 2

[48] Engin Tola, Vincent Lepetit, and Pascal Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *IEEE transactions on pattern analysis and machine intelligence*, 32(5):815–830, 2009. 2

[49] Prune Truong, Martin Danelljan, Luc Van Gool, and Radu Timofte. GOCor: Bringing globally optimized correspondence volumes into your neural network. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, 2020. 2

[50] Prune Truong, Martin Danelljan, and Radu Timofte. Glunet: Global-local universal network for dense flow and correspondences. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6258–6268, 2020. 1, 2

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, undefinedukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. 4

[52] Runzhong Wang, Junchi Yan, and Xiaokang Yang. Learning combinatorial embedding networks for deep graph matching.

In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3056–3065, 2019. 2

[53] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010. 2, 5, 6, 7

[54] Paul Wohlhart and Vincent Lepetit. Learning descriptors for object recognition and 3d pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3109–3118, 2015. 1

[55] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. 6

[56] Xu Yang, Zhi-Yong Liu, and Hong Qiao. A continuation method for graph matching based feature correspondence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(8):1809–1822, 2020. 2

[57] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 818–833, Cham, 2014. Springer International Publishing. 2

[58] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2

[59] Dong Zhang, Hanwang Zhang, Jinhui Tang, Meng Wang, Xiansheng Hua, and Qianru Sun. Feature pyramid transformer. In *European Conference on Computer Vision*, pages 323–339. Springer, 2020. 4

[60] Yinda Zhang, Sameh Khamis, Christoph Rhemann, Julien Valentin, Adarsh Kowdle, Vladimir Tankovich, Michael Schoenberg, Shahram Izadi, Thomas Funkhouser, and Sean Fanello. Activestereonet: End-to-end self-supervised learning for active stereo systems. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 784–801, 2018. 1

[61] Qijie Zhao, Tao Sheng, Yongtao Wang, Zhi Tang, Ying Chen, Ling Cai, and Haibin Ling. M2det: A single-shot object detector based on multi-level feature pyramid network. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 9259–9266, 2019. 3

[62] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021. 4