Weakly supervised learning of deep metrics for stereo reconstruction

Stepan Tulyakov and Anton Ivanov
École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
{stepan.tulyakov, anton.ivanov}@epfl.ch

Francois Fleuret
Idiap Research Institute, Switzerland
francois.fleuret@idiap.ch

Abstract

Deep-learning metrics have recently demonstrated extremely good performance to match image patches for stereo reconstruction. However, training such metrics requires large amount of labeled stereo images, which can be difficult or costly to collect for certain applications (consider, for example, satellite stereo imaging).

The main contribution of our work is a new weakly supervised method for learning deep metrics from unlabeled stereo images, given coarse information about the scenes and the optical system. Our method alternatively optimizes the metric with a standard stochastic gradient descent, and applies stereo constraints to regularize its prediction.

Experiments on reference data-sets show that, for a given network architecture, training with this new method without ground-truth produces a metric with performance as good as state-of-the-art baselines trained with the said ground-truth.

This work has three practical implications. Firstly, it helps to overcome limitations of training sets, in particular noisy ground truth. Secondly it allows to use much more training data during learning. Thirdly, it allows to tune deep metric for a particular stereo system, even if ground truth is not available.

1. Introduction

The stereo reconstruction problem consists in estimating a depth map from two images taken from different viewpoints. The problem has many practical applications in robotics [35], remote sensing [44], and 3D graphics [49].

It has been heavily investigated for several decades [41], and recent developments focused on designing high-order, region-based and object-specific priors [63, 10, 58, 18, 25, 30, 55, 54], and improving efficiency of large scale stereo [37, 26, 17, 7]. Perhaps the most significant recent breakthrough was to use deep metrics [12, 61]. It led to considerable gains in processing speed and reconstruction accuracy (see Tables 4, 5, and 6). Our work improves upon this line of research.

Main contributions. In this work we showed that it is possible to learn a high quality deep metric for stereo matching without any labeled training data, using only task-specific constraints. The method does not use the labeled training data but nevertheless relies on the information about the scenes and the optical system, hence our calling "weakly supervised". The method allows to: (1) learn a deep metric when the labeled training data is not available or contaminated with a noise; (2) use more data for the training and thus learn more complex, better performing networks while avoiding overfitting.

2. Related work

Stereo reconstruction algorithms rely on *epipolar geometry* [19], according to which every no-occluded point in one stereo view corresponds a point in the other view lying on a line that does not depend on the scene, but only on the optical system. This line is called an *epipolar line*, and for a *calibrated* stereo system, it is known for every image point. Furthermore, for a pinhole camera, all the points lying on a given epipolar line in the second view correspond to points lying on a common epipolar line in the first view. Such two epipolar lines are called *conjugate*.

It is a standard procedure to warp stereo views in order to make conjugate epipolar lines in these views horizontal and vertically aligned. This is called *stereo rectification*, and in a rectified stereo pair, every point from the first view corresponds to a point shifted horizontally in the second view. The extension of this shift – also known as a *disparity* – allows to compute the distance to the corresponding 3d point, which is the ultimate goal of the stereo reconstruction.

So at the core of the stereo reconstruction process lies the matching of similar patches in two images along epipolar lines and the estimation of the disparity. It is not a trivial task, since the local appearance of a physical point in the two views might differ due to radiometric and geometric distortions. The patch matching is usually performed using *invariant similarity measures* and *descriptors*, also known as features. Historically, the former were more popular for the stereo reconstruction, while the latter were used for matching sparse points of interest.

2.1. Similarity measures

The invariant similarity measures [22, 20] are popular for stereo reconstruction, probably due to their low computational complexity. The simplest similarity measures are the sum of absolute differences (SAD), and the sum of squared differences (SSD). Zero-mean variants of these methods (ZSAD, ZSSD), as well as sum of absolute gradient differences (GSAD), are invariant to local brightness changes, which can also be achieved by combining SAD and SSD with background subtraction by mean, Laplacian of Gaussian (LoG) [21] or Bilateral filters [4]. Non-parametric similarity measures, such as Rank and Census [59] are invariant to arbitrary order-preserving local intensity transformations, and measures such as the Mutual Information (MI) [24] explicitly model the joint intensity distribution in the two images, and are invariant to arbitrary intensity transformations. All these methods are invariant to radiometric distortions only.

2.2. Descriptors

Invariant descriptors are popular for sparse point matching, and are designed to be invariant to both radiometric and geometric distortions. They all are either local histograms of oriented image gradients such as SIFT [31], or binary strings of local pairwise pixel comparisons such as BRIEF [9]. Although descriptors are rarely used for stereo, there are some exceptions, such as DAISY [51], which can be efficiently computed densely.

Recently, the community has moved from these fully hand-crafted descriptors to data-driven descriptors, incorporating machine-learning approaches. Most of such descriptors perform discriminative dimensionality reduction either by feature selection, as VGG [46], linear feature extraction, as LDAHash [48], or boosting, as BinBoost [53].

2.3. Deep metrics

As for other application domains of machine learning, the current trend is to move beyond “shallow” models, where the learned quantities interact linearly with hand-designed non-linearities, but are not involved in further recombinations.

The resulting “deep metrics” demonstrate extremely good performance compared to other similarity measures and descriptors both for sparse point matching [23, 14, 45, 60, 57] and stereo reconstruction [61, 12].

Standard deep metric networks have a Siamese architecture, introduced in [8]. They consist of two “embedding” sub-networks with complete weight sharing that join into a common “head”. Each embedding sub-network is convolutional, it takes an image patch as input, and outputs the patch’s descriptor. The “head” is usually fully connected, it takes the two descriptors as input, and outputs a similarity measure. The Siamese architecture was firstly used for image patch matching in its classic form in [23]. Later it was shown, that the “head” network may be replaced by a fixed similarity such as L^2 [45] or cosine [61], that the embedding sub-networks may not share weights [60], and, finally, that the explicit notion of a descriptor might not be necessary [60].

2.4. Supervised learning of deep metrics

Existing methods for training a Siamese network for patch matching are supervised, using a training set composed of positive and negative examples. Each positive example (respectively negative) is a pair composed of a reference patch and its matching patch (respectively a non-matching one) from another image.

Training either takes one example at the time, positive or negative, and adapts the similarity [45, 12, 23, 60, 57], or takes at each step both a positive and a negative example, and maximizes the difference between the similarities, hence aiming at making the two patches from the positive pair “more similar” than the two patches from the negative pair [61, 27, 6]. This latter scheme is known as “Triplet Contrastive learning.”

Although the supervised learning of deep metrics works very well, the complexity of the models requires very large labeled training sets which are hard or costly to collect for real applications (consider, for example, our domain of interest – Mars satellite stereo reconstruction). Beside, even when such large sets are available, the ground truth is produced automatically from depth sensors and often contains noise that reduces effectiveness of the supervised learning [50] (please, refer supplementary materials for details). This can be mitigated by augmenting the training set with random perturbations [61] or synthetic training data [14, 34]. However, synthesis procedures are hand-crafted and do not account for the regularities specific to the stereo system and target scene at hand.

2.5. Weakly supervised learning

Our work is inspired by Multi-Instance Learning (MIL) [5] and Self-Training [52]. The main idea behind MIL, is to use “coarsely” labeled data, where one label indicates if a group of samples contains at least one positive sample. This allows to deal with low geometrical accuracy, or even the absence of geometrical information and a labeling at the scene level. It has been applied with success to deep learning [56].

Another strategy to relax the requirement for detailed labeling is Self-Training, where the training set is enriched with unlabeled data. As for transductive learning, Self-Training works by leveraging the information carried by the unlabeled data about the structure of the data population [11, 38]. Note, that in contrast to Self-Training, which is a semi-supervised method utilizing mixture of the labeled and the unlabeled training data, our method is a weakly supervised method, because it uses only unlabeled training data and the prior knowledge about the structure of the data population in a form of constraints as in [47].

Our most efficient method uses dynamic programming (DP) to regularize the noisy prediction of the metric as it is currently trained. Similar idea in a different context appeared in [28]. In both [28] and in our paper the high level idea is to learn an embedding in a weakly supervised manner by minimizing the energy of the best path on a constraint graph, while simultaneously maximizing the energy of the best unconstrained path. However there are important differences: (1) we use per location loss function with margin, while they use per path loss function without margin, (2) we use stereo constraints to construct the constrained graph, while they use text string labels, and finally (3) the application domain is drastically difference since we deal with geometrical regression and they deal with classification. DP has also been used to automatically segment sequences of action demonstrations into macro-actions to deal with non-Markovian decision processes [29], and the k-shortest paths algorithm, which is a generalization of dynamic programming to multiple paths, was used to train a person detector from videos with time-sparse ground-truth [3].

Our work is also close to [15], where unsupervised learning is used to train regression CNN for predicting depth from a single image. Although regularization and alternating procedure play a central role in both our work and theirs, objectives and losses differ (regression vs. patch metric, reconstruction vs. classification).

3. Method

We start by formulating in § 3.1 the task of weakly supervised deep metric learning for stereo, then in § 3.2 we review the stereo matching problem constraints we consider, and in § 3.3 we describe how we use them to drive the training.

3.1. Problem formulation

We are provided with a weakly supervised training set $\mathbf{Tr} = \{(\mathbf{e}^r, \mathbf{e}^+, \mathbf{e}^-)_n\}_{n=1:N}$. Each training example is a triplet of series of $s \times s$ gray-scale patches:

- *reference patches* $\mathbf{e}^r = (p^r_1, p^r_2, .., p^r_W)$ extracted from a horizontal line of a left rectified stereo image,
- *positive patches* $\mathbf{e}^+ = (p^+_1, p^+_2, .., p^+_W)$ extracted from the corresponding horizontal in the right rectified stereo image, and
- *negative patches* $\mathbf{e}^- = (p^-_1, p^-_2, .., p^-_W)$ extracted from another horizontal line of a right rectified stereo image,

where W is the number of patches per line, and N is the number of training examples. In addition to the training set, we are provided with the maximum possible disparity d_{max}, which depends on the optical system and a prior knowledge about the scene.

Our goal is to learn parameters $\mathbf{\Theta}$ of deep metric $S(x, y, \mathbf{\Theta})$ such that, for any set of reference $\mathbf{e}^r$ and positive image patches $\mathbf{e}^+$, the row-wise maxima of the *similarity matrix* $\mathbf{S}^{r+}_{ij} = S\left(p^r_i, p^+_j, \mathbf{\Theta}\right)$ correspond to the true matches.

Note, that in contrast to [23, 60, 45, 14, 57, 12, 61] in our case each training example is not a pair of patches, but a triplet of series of patches each taken on a horizontal line of a rectified stereo image, so that we can utilize constraints and loss functions defined on such families of patches jointly. Additionally, processing lines as a whole significantly speeds up the training process by allowing to reuse shared computations.

3.2. Matching constraints

The stereo matching problem satisfies the following constraints:

(E) **Epipolar constraint.** Every non-occluded reference patch has a matching positive patch [19][239-241p].

(D) **Disparity range constraint.** The offset of the reference patch index with respect to the matching positive patch index is bounded by a maximum disparity d_{max}. This comes from the stereo system parameters (focal length, pixel size, baseline) and the distance range of the scenes.

(U) **Uniqueness constraint.** The matching positive patch is unique [33].

(C) **Continuity (smoothness) constraint.** The offsets of the reference patches indices with respect to the matching positive patch indices are similar for nearby reference patches everywhere except on depth discontinuities [33].

(O) **Ordering constraint.** The reference patches are ordered on their lines as the matching positive patches on theirs.

These constraints result in a particular shape of the positive similarity matrix, as pictured in Figure 1. Note that uniqueness (U), continuity (C) and ordering (O) constraints are sometimes violated. However, experiments show that these rare violations only marginally affect the training in presence of large training set.

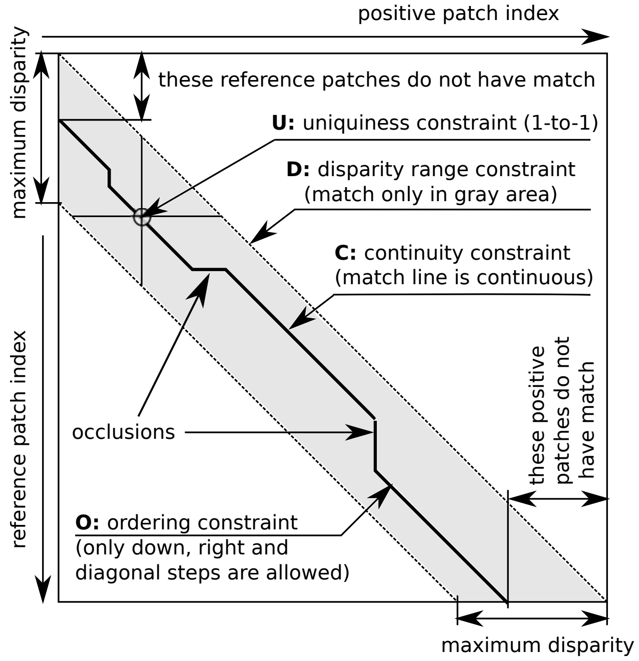
positive patch index
maximum disparity
these reference patches do not have match
U: uniqueness constraint (1-to-1)
D: disparity range constraint
(match only in gray area)
C: continuity constraint
(match line is continuous)
reference patch index
occlusions
these positive patches do not have match
O: ordering constraint
(only down, right and diagonal steps are allowed)
maximum disparity

Figure 1. Positive similarity matrix. The bold line corresponds to the optimal matches that satisfy the stereo constraints. Elements within the disparity range are shown in gray. Note that there are no matches for some points on the reference and positive epipolar lines.

3.3. Proposed weakly supervised methods

We developed several weakly supervised methods that use different subsets of the stereo constraints during training. All methods alternate between two steps: (2) compute constrained and unconstrained matches, given the current estimate of the metric, (1) improve the metric, given the current estimate of the matches. They can be used in combination with any deep metric architecture and any gradient based optimization method.

To each of our methods corresponds a loss function optimized in each of the two steps mentioned above. It takes as an input either $\mathbf{S}^{r+}$, or the three matrices $\mathbf{S}^{r+}$, $\mathbf{S}^{r-}$ and $\mathbf{S}^{-+}$ defined respectively as follows:

$$S^{r+}_{ij} = \begin{cases} S(p^r_i, p^+_j, \mathbf{\Theta}) & 0 \le i - j \le d_{max} \\ -\infty & otherwise \end{cases} \qquad (1)$$

$$S^{r-}_{ij} = \begin{cases} S(p^r_i, p^-_j, \mathbf{\Theta}) & 0 \le i - j \le d_{max} \\ -\infty & otherwise \end{cases} \qquad (2)$$

$$S^{-+}_{ij} = \begin{cases} S(p^-_i, p^+_j, \mathbf{\Theta}) & 0 \le i - j \le d_{max} \\ -\infty & otherwise \end{cases} \qquad (3)$$

In the next sections we describe each method in details.

3.3.1 MIL method

This method is inspired by Multi-Instance Learning (MIL) paradigm [5] and uses only the epipolar and the disparity range constraints (E) and (D) from § 3.2.

From these two constraints, we know that every non-occluded reference patch has a matching positive patch in a known index interval, but does not have a matching negative patch. Therefore, for every reference patch, the similarity of the best reference-positive match should be greater than similarity of the best reference-negative match. Our training objective is to push apart these two similarities.

The training loss for the MIL method is

$$L(\mathbf{\Theta}) = \frac{1}{|\mathbf{rows}|} \sum_{i \in \mathbf{rows}} \max(0, -\max_j S^{r+}_{ij} + \max_j S^{r-}_{ij} + \mu)$$
$$+ \frac{1}{|\mathbf{cols}|} \sum_{j \in \mathbf{cols}} \max(0, -\max_i S^{r+}_{ij} + \max_i S^{-+}_{ij} + \mu), \qquad (4)$$

where $\mathbf{rows} = \{d_{max} + 1, \dots, W\}$ is a set of rows of the similarity matrix that are guaranteed to have correct matches (see Fig 1), $\mathbf{col} = \{1, \dots, W - d_{max}\}$ is a set of valid columns of the similarity matrix that are guaranteed to have correct matches, W is the number of patches in a horizontal line of rectified image, and μ is a loss margin. Note that the disparity range constraint is taken into account automatically, if we use the similarity matrices as defined in § 3.3.

Experiments shows that the method learns metrics insensitive to small shifts from the optimal match. This problem results in blocky shape of a similarity matrix, where blocks correspond to the areas where the metric is not able to find unique match. This issue motivates the CONTRASTIVE method described in the following section.

3.3.2 CONTRASTIVE method

This method uses the epipolar, the disparity range, and the uniqueness constraints (E), (D), and (U) from § 3.2.

From the epipolar and the disparity range constraints we know that every non-occluded reference patch has a matching positive patch in a known index interval. Furthermore, according to the uniqueness constraint the matching positive patch is unique. Therefore, for every patch, the similarity of the best match should be greater than the similarity of the second best match. Our training objective is to push apart these two quantities.

The training loss for this CONTRASTIVE method is

$$L(\mathbf{\Theta}) = \frac{1}{|\mathbf{rows}|} \sum_{i \in \mathbf{rows}} \max(0, -\max_j S^{r+}_{ij} + \max_j \hat{S}^{r+}_{ij} + \mu)$$
$$+ \frac{1}{|\mathbf{cols}|} \sum_{j \in \mathbf{cols}} \max(0, -\max_i S^{r+}_{ij} + \max_i \check{S}^{r+}_{ij} + \mu), \qquad (5)$$

where $\hat{\mathbf{S}}$ is a similarity matrix with masked out row-wise maxima, $\check{\mathbf{S}}$ is a similarity matrix with masked out columnwise maxima. To mask out elements of similarity of matrix, we simply substitute them with $-\infty$.

Training with this method and CONTRASTIVE-DP method, from § 3.3.3, requires good metric initialization. Fortunately, in this work we use deep metric that works very well with random CNN weights, as shown in § 4.7, thus we were able to use contrastive methods right from the start. In other cases, it would be necessary to pre-train a metric with another method such as MIL from § 3.3.1.

Experiments show that the CONTRASTIVE method suffers from a problem opposite to the one exhibited by the MIL method: it produces over-sharpened metric, sensitive even to small shifts from the exact match. This is also detrimental to the performance, since our goal is to find metric invariant to small geometric transformations, such as shift. We solved the problem by masking out all spatial neighbors within t_{sup} radius from the maxima in $\hat{\mathbf{S}}$ and in $\check{\mathbf{S}}$. See the supplementary materials for details.

3.3.3 CONTRASTIVE-DP method

This method uses all constraints listed in § 3.2. The only difference with CONTRASTIVE is that it finds the best match under (C) and (O) using dynamic programming (DP), instead of independent maxima.

Formally, it solves

$$p^* = \operatorname*{argmax}_{p \in \mathcal{P}} \frac{1}{|p|} \sum_{(i,j) \in p} \mathbf{S}^{r+}_{ij}, \tag{6}$$

where $\mathcal{P}$ is the set of paths $\{(i_n, j_n)\}_{n=1:M}$ which are continuous in the following sense:

$$\forall n > 1, (i_n, j_n) - (i_{n-1}, j_{n-1}) \in \{(0,1), (1,0), (1,1)\},$$
$$\text{and } (i_1, j_1) \in \{1\} \times [1, d_{max}].$$

Which means that only down, right and diagonal steps are allowed. This enforces the continuity and the ordering constraints (C) and (O) in the solution. Notice also that we search for a path that has maximum average energy rather than maximum total energy to prevent a bias toward longer paths and consequently smaller disparities.

Given the best match-path p^* found by the dynamic programming we define our loss function as

$$\begin{aligned} L(\mathbf{\Theta}) = & \frac{1}{|p^*|} \sum_{(i,j) \in p^*} \max(0, -S^{r+}_{ij} + \max_{k} \tilde{S}^{r+}_{ik} + \mu) + \\ & \frac{1}{|p^*|} \sum_{(i,j) \in p^*} \max(0, -S^{r+}_{ij} + \max_{l} \tilde{S}^{r+}_{lj} + \mu), \end{aligned} \tag{7}$$

where $\tilde{\mathbf{S}}$ is a similarity matrix where all neighbors of elements belonging to p^* withing radius are masked out by setting their values to $-\infty$.

The best match-path computed by the dynamic programming might contain vertical and horizontal segments, that correspond to depth discontinuities, violating the uniqueness constraint (U). Therefore, during the training we ignore all such segments that are longer than t_{occ}. For more details, please refer to the supplementary materials.

4. Experiments

Our experiments were done in the Torch framework [13]. Optimization was performed with the ADAM method with standard settings, using mini-batches of size equal to the training images height, and no data augmentation of any sort. The initialization of weights and biases of our deep metric network was done in standard way by random sampling from zero-mean uniform distribution.

We guarantee reproducibility of all experiments in this section by using only available data-sets, and making our code available online under open-source license after publications.

4.1. Data-Sets

In our experiments we use three popular benchmark data-sets: KITTI'12 [16], KITTI'15 [35] and Middlebury (MB) [41, 42, 40, 22, 39]. These data-sets have online scoreboards [1, 2], showing comparative performance of all participating stereo methods.

KITTI'12 and KITTI'15 data-sets each consist of 200 training and 200 test rectified stereo pairs of resolution 1226×370 acquired from cars moving around a city. About 30% of the pixels in the training set are supplied with a ground truth disparity acquired by a laser altimeter with error less than 3 pixels. The disparity range is about 230 pixels. Each data-set is supplied with an extension (respectively KITTI'12-EXT and KITTI'15-EXT) that contains 19 additional stereo pairs for each scene, without ground truth disparity. This allows us to use 40× more training data for the weakly supervised learning than for the supervised (actually even more, considering that only about 30% of pixels in the training set have labels).

Middlebury data-set (MB) consists of 60 training and 30 test rectified stereo pairs. The images are acquired by different stereo systems and contain different artificial scenes. Their resolution varies from 380×430 to 3000×2000, and their disparity ranges vary from 30 to 800 pixels. The training images are provided with a dense ground truth disparity acquired by structured light system with error less that 0.2 pixels.

4.2. Performance measure

To estimate the performance of deep metrics we compute a prediction error rate defined as the proportion of non-occluded patches for which the predicted disparity is off by more than 3 pixels.

The motivation behind this work is to improve the metric as a mean to match patches in a stand-alone manner, as we have not taken into account the interplay with the additional post-processing that may be applied in a complete stereo pipeline. Performance regarding this main objective is measured by picking the patch with the largest similarity among the patches that belong to a valid disparity range on the epipolar line. We call this the *winner-take all* (WTA) error rate.

A second measure is the error rate of a complete stereo pipeline with plugged-in deep metric. This is a performance measure of direct practical interest, although not the objective we optimize during our training.

4.3. Deep metric architecture

The main contribution of this work is a new weakly supervised training method, not deep metric architecture, therefore we simply adopt the overall architecture of well performing MC-CNN fst network from [61], shown in Table 1, and substitute their learning method with ours.

Parameter	**KITTI'12,15**	**MB**
Number of CNN layers	4	5
Number of features per layer	64	64
Receptive field	3x3x64	3x3x64
Activation function	ReLU	ReLU
Equivalent patch size	9x9	11x11
Similarity metric	Cosine	Cosine

Table 1. Network architectures for deep metric from [61] that we use in our experiments.

4.4. Comparison of weakly supervised methods

In this experiment we compare the performance of the proposed weakly supervised methods. We performed comparison on KITTI'12 data-set using the winner-take-all (WTA) error (see § 4.2). The results of the experiments are shown in Table 2.

Method	**WTA error, [%]**	**Time, [hr]**
MIL	18.45	45
CONTRASTIVE	17.63	30
MIL+CONTRASTIVE	16.12	65
CONTRASTIVE-DP	**14.61**	**68**

Table 2. Comparison of the proposed weakly supervised learning methods on KITTI'12 set. MIL+CONTRASTIVE method uses MIL and CONTRASTIVE losses simultaneously. All methods are used to train the same network architecture. The CONTRASTIVE-DP method, shown in bold, uses all the constraints during learning and achieves the smallest WTA error. Notice that in general increasing the number of constraints increases performance.

The main conclusion is that weakly supervised methods that use more stereo constraints during learning perform better. For example, the MIL, that uses only the epipolar and the disparity range constraints, has larges WTA error, whereas the CONTRASTIVE-DP, that uses the epipolar, the disparity range, the continuity, the uniqueness and the ordering constraints has smallest WTA error.

In all following sections, we use the best performing CONTRASTIVE-DP method only, and refer to it as MC-CNN-WS, where WS stands for weakly supervised.

4.5. Comparison with supervised method

In this section, we compare the proposed weakly supervised method with our reference fully supervised deep-metric baseline [61] on the three different sets, using the winner-take-all (WTA) error (see § 4.2).

The results are shown in Table 3. As we see, our method outperforms the supervised method in terms of WTA error across tree sets. This is remarkable considering the fact that our method does not use ground truth disparity during learning.

The success of our method in case of KITTI'12 and KITTI'15 sets can be attributed to the fact that these sets have large amount of unlabeled stereo data, that can be used by our method. In fact, these sets have more than 40× more unlabeled data than labeled training data.

In case of MB data-set our method does not have such huge advantage over the supervised method. The set has only 30% more unlabeled training data than the labeled training data.

Method	**WTA error, [%]**		
	KITTI'12	**KITTI'15**	**MB**
MC-CNN fst [61]	15.44	15.38	29.94
MC-CNN-WS fst (ours)	**13.90**	**14.08**	**29.60**
CENSUS 9x9 [59]	53.52	50.35	64.53
SAD 9x9	32.36	30.67	59.39

Table 3. Comparison of our weakly supervised learning method with the fully supervised baseline using the same network architecture [61]. Smallest WTA errors are shown in bold. Our weakly supervised method outperforms the baseline in terms of WTA error across tree sets. This is remarkable since in contrast to the supervised method, our does not use ground truth disparity during learning. For reference, the two bottom rows show the performance of two standard similarity measures and descriptors, where SAD stands for sum of absolute differences of pixels' intensities in 9×9 image patch. Note that following the setup of [61], the patches used as input to the deep-learning methods are of size 9×9 for KITTI'12,'15, and 11×11 for MB.

4.6. Stereo benchmarking

In this section we investigate how well our weakly supervised deep metric performs when it is combined with the complete stereo pipeline. For that we plug it in the stereo pipeline from [61], and tuned the parameters of the pipeline using simple coordinate descent method, starting from the default values of [61]. Note that we used specific metric and pipeline parameters for each data-set.

Then we computed disparity maps for the test sets with withheld ground truth, and uploaded the results to the evaluation web sites for the respective data-sets[1, 2]. The obtained evaluation results are available in online scoreboards and shown in Tables 5, 6 and 4 (note, that corresponding disparity maps are also available for viewing in the scoreboards). As we can see, results with our metric trained without ground truth during training are very close to the results of the fully supervised method across all benchmarks.

Those are very encouraging results, given in particular that we did not optimize the deep metric and the pipeline parameters together, and considering the performance in the winner-take-all setup of § 4.5. The fact that our metric outperforms the supervised metric in winner-takes-all setup but lags behind it when used as a part of the Pipeline is not surprising. The pipeline relies on multiple heuristics and as such provide a regularization that is not taken into account during the training of the network. Thus, smaller WTA error does not guarantee smaller Pipeline error.

Regarding the processing time, note that the network structure used for our method is identical to that of MC-CNN-fst [61], except for the pipeline parameters. The difference in processing times in Tables 5, 6 and 4 is only due to the hardware differences.

#	dd/mm/yy	Algorithm	Pip. Err, [%]	Time, [s]
1	19/01/15	NTDE [25]	7.62	300
2	28/08/15	MC-CNN acrt [61]	8.29	254
3	03/11/15	MC-CNN+RBS [7]	8.62	345
4	**26/01/16**	**MC-CNN fst [61]**	**9.69**	**2.94**
5	**11/14/16**	**MC-CNN-WS (ours)**	**12.3**	**5.59**
6	13/10/15	MDP [30]	12.6	130
7	19/04/15	MeshStereo [63]	13.4	146

Table 4. MB benchmark [2] snapshot from 14/11/2016 with published methods (default view). Methods ranked 1, 2, 3, 4 and 5 use deep metrics for stereo matching. Note that our weakly supervised method MC-CNN-WS, shown in bold, that does not use ground truth data during training, has an error rate very similar to that of the supervised MC-CNN fst method, also shown in bold, trained with ground truth data.

4.7. What does deep metric learn?

In Figure 2 we show positive similarity matrices computed by the network initialized with random weight and the network after the training with MC-CNN-WS on KITTI'12 data-set. While one can not visually distinguish the best match in the similarity matrices before the training, it becomes clearly visible after. This suggests that the training improves discriminative ability of the deep metric.

#	dd/mm/yy	Algorithm	Pip. Err, [%]	Time, [s]
1	27/04/16	PBCP [43]	2.36	68
2	26/10/15	Displets v2 [18]	2.37	265
3	21/08/15	MC-CNN acrt [61]	2.43	67
4	30/03/16	cfusion [36]	2.46	70
5	16/04/15	PRSM [55]	2.78	300
6	**21/08/15**	**MC-CNN fst [61]**	**2.82**	**0.8**
7	03/08/15	SPS-st [58]	2.83	2
8	**14/11/16**	**MC-CNN-WS (ours)**	**3.02**	**1.35**
9	03/03/14	VC-SF [54]	3.05	300

Table 5. KITTI'12 benchmark [1] snapshot from 14/11/2016 with published methods (default view). Methods ranked 1, 2, 3, 4, 6, and 7 use deep metrics for stereo matching. Note that our weakly supervised method MC-CNN-WS, shown in bold, that does not use ground truth data during training, has an error rate very similar to that of the supervised MC-CNN fst method, also shown in bold, trained with ground truth data. Since the MC-CNN fst method does not appear on KITTI'12 evaluation table, due to restrictions on the number of results for a single paper, we borrowed it from [62]

#	dd/mm/yy	Algorithm	Pip. Err, [%]	Time, [s]
1	26/10/15	Displets v2 [18]	3.43	265
2	27/04/16	PBCP [43]	3.61	68
3	21/08/15	MC-CNN acrt [61]	3.89	2.94
4	16/04/15	PRSM [55]	4.27	300
5	06/11/15	DispNetC [34]	4.34	0.06
6	11/04/16	ContentCNN [32]	4.54	1
7	**21/08/15**	**MC-CNN fst [61]**	**4.62**	**0.8**
8	**14/11/16**	**MC-CNN-WS (ours)**	**4.97**	**1.35**
9	03/08/15	SPS-st [58]	5.31	2

Table 6. KITTI'15 benchmark [1] snapshot from 14/11/2016 with published methods (default view). Methods ranked 1, 2, 3, 5, 6 and 7 use deep metrics for stereo matching. Note that our weakly supervised method MC-CNN-WS, shown in bold, that does not use ground truth data during training, has an error rate very similar to that of the supervised MC-CNN fst method, also shown in bold, trained with ground truth data. Since the MC-CNN fst method does not appear on KITTI'12 evaluation table, due to restrictions on the number of results for a single paper, we borrowed it from [62].

Notably, the performance of the deep metric with a random weights is surprisingly good. The corresponding WTA error on KITTI'12 is just 42.01%. This good performance is the reason why we don't need to pre-train the deep metric before applying our contrastive methods.

In Figure 3 we show failure cases of learned deep metric. Most of the failures happen when the ground truth match is visually indistinguishable from the incorrect match picked by the deep metric. This happens if the reference patch is from a flat image area, an area with a repetitive texture, or an area with a horizontal edge.

Notably, some failures are triggered by probable errors in the ground truth. These errors worsen outcomes of the supervised learning as we show in supplementary materials, but does not affect outcomes of our weakly supervised

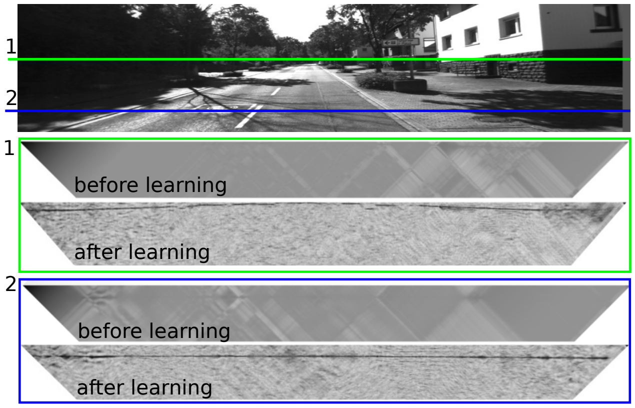

Figure 2. Diagonal part of the similarity matrix before and after training with MC-CNN-WS on KITTI'12 dataset. Top figure shows one of the stereo images with two highlighted epipolar lines. The pictures below show the positive similarity matrices for these epipolar lines. The dark elements in the similarity matrices correspond to the higher similarities. WTA error before training is 42.01%, and 14.61% after. Note that before the training we can not visually distinguish the best matches in the similarity matrices, while after the learning they are clearly visible.

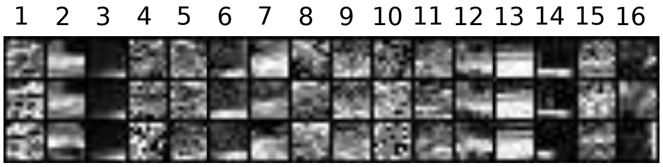

Figure 3. Failure cases of the deep metric trained with our MC-CNN-WS method on the KITTI data-set. For each example the three patches displayed correspond to (from top to bottom): the reference patch, the predicted match and the ground-truth match. Note that as expected, the ground truth and the predicted matches are often visually indistinguishable. This happens if the reference patch is from an area with almost horizontal edges (3, 6, 13), a flat image area (4, 5, 10), or an area with repetitive texture. Some failures are triggered by likely errors in the ground truth labeling (2, 12, 14, 16).

learning, since it does not use the ground-truth.

5. Conclusion

We proposed novel weakly supervised techniques for training patch similarity measures for stereo reconstruction. These techniques allow to train with data-sets for which ground truth is not available, by relying on simple constraints coming from properties of the optical sensor, and from a rough knowledge about the scenes to process.

We applied this framework to the training of a "deep metric", that is a deep siamese neural-network that takes two patches as an input and predicts a similarity measure. Benchmarking on standard data-sets shows that the resulting performance is as good or better than published results with the same network trained on the same but fully labeled data-sets (see Table 3).

This very good performance can be explained by the strong redundancy of a fully labeled data-set, due to the continuity of surfaces, coupled with inevitable labeling errors. The latter can degrade the performance resulting from a fully supervised training process, and could only be mitigated by using a prior knowledge about the regularity of the labeling, similar to the constraints we use.

The techniques we propose open the way first to using stereo reconstruction based on deep metrics for data-sets for which no ground-truth exists, such as planetary measurements. Second, it will allow the training of larger neural networks, with very large unlabeled data-sets. Our experiments show that the network that we are using in our experiments does benefit from an one order of magnitude more training samples, than it is available to supervised method as shown in Table 3. We expect that this effect will be even more significant if we use our training method with larger networks that would over-fit existing labeled training sets.

We are now extending our work in two directions. Firstly, we are generalizing the CONTRASTIVE-DP method to use any stereo pipeline instead of DP to compute the regularized solution. This version can piggyback on any stereo pipeline, and tune the deep metric for the stereo system at hand, during normal operation, with minimum computational overhead. Secondly, we are investigating how the unlabeled training data can be combined with the labeled data in the framework of our algorithm. Our preliminary experiments show both directions to be very promising.

6. Acknowledgement

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40 GPU used for this research. We also acknowledge support from the NCCR PlanetS and CaSSIS project of the University of Bern funded through the Swiss Space Office via ESA's PRODEX programme.

References

[1] KITTI 2012, 2015 stereo scoreboards. `http://www.cvlibs.net/datasets/kitti/`. Accessed: 2016-11-14.

[2] Middlebury scoreboard. `http://vision.middlebury.edu/stereo/eval3/`. Accessed: 2016-11-14.

[3] K. All, D. Hasler, and F. Fleuret. FlowBoost - Appearance learning from sparsely annotated video. In *CVPR*, 2011.

[4] A. Ansar, A. Castano, and L. Matthies. Enhanced Real-time Stereo Using Bilateral Filtering . *3DPVT*, 2004.

[5] B. Babenko. Multiple instance learning: algorithms and applications. *NCBI Google Scholar*, 2008.

[6] V. Baltnas, E. Johns, L. Tang, and K. Mikolajczyk. PN-Net: Conjoined Triple Deep Network for Learning Local Image Descriptors. *CoRR*, 2016.

[7] J. T. Barron and B. Poole. The fast bilateral solver. *ECCV*, 2016.

[8] J. Bromley, I. Guyon, Y. Lecun, E. Sckinger, and R. Shah. Signature verification using a "siamese" time delay neural network. In *NIPS*, 1994.

[9] M. Calonder, V. Lepetit, M. Özuysal, T. Trzcinski, C. Strecha, and P. Fua. BRIEF: Computing a local binary descriptor very fast. *PAMI*, 2012.

[10] A. Chakrabarti, Y. Xiong, S. J. Gortler, and T. Zickler. Low-Level Vision by Consensus in a Spatial Hierarchy of Regions. *CVPR*, 2015.

[11] X. Chen, A. Shrivastava, and A. Gupta. NEIL: Extracting Visual Knowledge from Web Data. In *ICCV*, 2013.

[12] Z. Chen, X. Sun, and L. Wang. A Deep Visual Correspondence Embedding Model for Stereo Matching Costs. *ICCV*, 2015.

[13] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, 2011.

[14] P. Fischer, A. Dosovitskiy, and T. Brox. Descriptor Matching with Convolutional Neural Networks: a Comparison to SIFT. *CoRR*, 2014.

[15] R. Garg, V. K. B. G, and I. D. Reid. Unsupervised CNN for single view depth estimation: Geometry to the rescue. *ECCV*, 2016.

[16] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012.

[17] A. Geiger, M. Roser, and R. Urtasun. Efficient large-scale stereo matching. *ACCV*, 2010.

[18] F. Güney and A. Geiger. Displets: Resolving Stereo Ambiguities using Object Knowledge. *CVPR*, 2015.

[19] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.

[20] H. Hirschm. Evaluation of Stereo Matching Costs on Images with Radiometric Differences. *PAMI*, 2008.

[21] H. Hirschmüller, P. R. Innocent, and J. Garibaldi. Real-Time Correlation-Based Stereo Vision with Reduced Border Errors. *IJCV*, 2002.

[22] H. Hirschmuller and D. Scharstein. Evaluation of Cost Functions for Stereo Matching. *CVPR*, 2007.

[23] M. Jahrer, M. Grabner, and H. Bischof. Learned local descriptors for recognition and matching. *Computer Vision Winter Workshop*, 2008.

[24] J. Kim, V. Kolmogorov, and R. Zabih. Visual correspondence using energy minimization and mutual information. In *ICCV*, 2003.

[25] K. R. Kim and C. S. Kim. Adaptive smoothness constraints for efficient stereo matching using texture and edge information. In *ICIP*, 2016.

[26] J. Kowalczuk, E. T. Psota, and L. C. Pérez. Real-time Stereo Matching on CUDA using an Iterative Refinement Method for Adaptive Support-Weight Correspondences Real-time Stereo Matching on CUDA using an Iterative Refinement Method for Adaptive Support-Weight Correspondences. *Transactions on Circuits and Systems for Video Technology*, 2012.

[27] B. G. V. Kumar, G. Carneiro, and I. Reid. Learning Local Image Descriptors with Deep Siamese and Triplet Convolutional Networks by Minimising Global Loss Functions. *CVPR*, 2016.

[28] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

[29] L. Lefakis and F. Fleuret. Dynamic Programming Boosting for Discriminative Macro-Action Discovery. *ICML*, 2014.

[30] A. Li, D. Chen, Y. Liu, and Z. Yuan. Coordinating multiple disparity proposals for stereo computation. In *CVPR*, 2016.

[31] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 2004.

[32] W. Luo, A. G. Schwing, and R. Urtasun. Efficient deep learning for stereo matching. In *CVPR*, 2016.

[33] D. Marr and T. Poggio. A Computational Theory of Human Stereo Vision. *Biological Sciences*, 1979.

[34] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. *CVPR*, 2016.

[35] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *CVPR*, 2015.

[36] V. Ntouskos and F. Pirri. Confidence driven tgv fusion. *CoRR*, 2016.

[37] E. T. Psota, J. Kowalczuk, M. Mittek, and L. C. Perez. MAP Disparity Estimation Using Hidden Markov Trees. *ICCV*, 2015.

[38] S. E. Reed and H. Lee. Raining deep neural networks on noisy labels with bootstrapping. *ICLR*, 2015.

[39] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nešić, X. Wang, and P. Westling. High-resolution stereo datasets with subpixel-accurate ground truth. *Lecture Notes in Computer Science*, 2014.

[40] D. Scharstein and C. Pal. Learning conditional random fields for stereo. In *CVPR*, 2007.

[41] D. Scharstein and R. Szeliski. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. *IJCV*, 2001.

[42] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. *CVPR*, 2003.

[43] A. Seki and M. Pollefeys. Patch based confidence prediction for dense disparity map. In *BMVC*, 2016.

[44] D. E. Shean, O. Alexandrov, Z. M. Moratto, B. E. Smith, I. R. Joughin, C. Porter, and P. Morin. An automated, open-source pipeline for mass production of digital elevation models (DEMs) from very-high-resolution commercial stereo satellite imagery. {*ISPRS*}, 2016.

[45] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer. Discriminative Learning of Deep Convolutional Feature Point Descriptors. *ICCV*, 2015.

[46] K. Simonyan, A. Vedaldi, and A. Zisserman. Descriptor Learning Using Convex Optimization. *PAMI*, 2013.

[47] R. Stewart and S. Ermon. Label-Free Supervision of Neural Networks with Physics and Domain Knowledge. *AAAI*, 2017.

[48] C. Strecha, A. M. Bronstein, M. M. Bronstein, and P. Fua. LDAHash: Improved matching with smaller descriptors. *PAMI*, 2012.

[49] C. Strecha, T. Pylvänäinen, and P. Fua. Dynamic and scalable large scale image reconstruction. In *CVPR*, 2010.

[50] S. Sukhbaatar and R. Fergus. Learning from noisy labels with deep neural networks. *ICLR'15 workshop*, 2014.

[51] E. Tola. DAISY: A Fast Descriptor for Dense Wide Baseline Stereo and Multiview Reconstruction. *PAMI*, 2010.

[52] I. Triguero, S. García, and F. Herrera. Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. *Knowledge and Information Systems*, 2013.

[53] T. Trzcinski, M. Christoudias, V. Lepetit, and P. Fua. Learning Image Descriptors with the Boosting-Trick. *NIPS*, 2012.

[54] C. Vogel, S. Roth, and K. Schindler. View-consistent 3d scene flow estimation over multiple frames. In *ECCV*, 2014.

[55] C. Vogel, K. Schindler, and S. Roth. 3d scene flow estimation with a piecewise rigid scene model. *IJCV*, 2015.

[56] J. Wu, Y. Yu, C. Huang, and K. Yu. Deep Multiple Instance Learning for Image Classification and Auto-Annotation. *CVPR*, 2015.

[57] Xufeng Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg. MatchNet: Unifying feature and metric learning for patch-based matching. *CVPR*, 2015.

[58] K. Yamaguchi, D. Mcallester, and R. Urtasun. Efficient Joint Segmentation , Occlusion Labeling , Stereo and Flow Estimation. *ECCV*, 2014.

[59] R. Zabih and J. Woodfill. Non-parametric Local Transforms for Computing Visual Correspondence. *ECCV*, 1994.

[60] S. Zagoruko and N. Komodakis. Learning to Compare Image Patches via Convolutional Neural Networks. *CVPR*, 2015.

[61] J. Žbontar and Y. LeCun. Computing the Stereo Matching Cost With a Convolutional Neural Network. *CVPR*, 2015.

[62] J. Zbontar and Y. LeCun. Stereo matching by training a convolutional neural network to compare image patches. *JMLR*, 2016.

[63] C. Zhang and Z. Li. MeshStereo : A Global Stereo Model with Mesh Alignment Regularization for View Interpolation. *ICCV*, 2015.