TS-NET: COMBINING MODALITY SPECIFIC AND COMMON FEATURES FOR MULTIMODAL PATCH MATCHING

Sovann En, Alexis Lechervy, Frédéric Jurie

TS-NET: COMBINING MODALITY SPECIFIC AND COMMON FEATURES FOR MULTIMODAL PATCH MATCHING

Sovann En, Alexis Lechervy, Frédéric Jurie

Normandie Univ, UNICAEN, ENSICAEN, CNRS – UMR GREYC

ABSTRACT

Multimodal patch matching addresses the problem of finding the correspondences between image patches from two different modalities, *e.g.* RGB vs sketch or RGB vs near-infrared. The comparison of patches of different modalities can be done by discovering the information common to both modalities (Siamese like approaches) or the modality-specific information (Pseudo-Siamese like approaches). We observed that none of these two scenarios is optimal. This motivates us to propose a three-stream architecture, dubbed as TS-Net, combining the benefits of the two. In addition, we show that adding extra constraints in the intermediate layers of such networks further boosts the performance. Experimentations on three multimodal datasets show significant performance gains in comparison with Siamese and Pseudo-Siamese networks[†].

Index Terms— Multimodal Patch Matching, Siamese network, Deep Metric Learning

1. INTRODUCTION AND RELATED WORK

Patch matching, the task consisting in determining the correspondences between image patches, is essential for many computer vision problems, *i.e.*, multi-view reconstruction, structure from motion, object-instance recognition, *etc*. In this work, we aim to study the problem of matching patches in a multimodal setting where input patches come from different sources, *i.e.* RGB images vs hand-drawn sketches or RGB vs near-infrared images.

Broadly speaking, there are two main ways to design local patch matching systems, either by employing handcrafted features or through machine learning techniques. Pioneer works in patch matching [1] are based on handcrafted features such as the SIFT descriptor/detector or some variants, *e.g.* [2], DAISY, [3], *etc.* Such approaches usually use conventional distance to measure patch similarity, *e.g.* the Euclidean distance, which usually does not provide an optimal solution for matching purposes. This family of approaches relies heavily on human expertise.

In contrast with feature engineering, another approach to patch matching consists in using supervised algorithms to find

---

adapted features or adapted similarity functions, for given datasets. Machine learning allows to find optimal projections minimizing (or maximizing) the distances between positive patches (negative patches respectively) [4, 5, 6].

Recent breakthroughs in deep learning have strongly contributed to this field. One of the first works in deep metric learning is the one of Jahrer *et al.* [7] introducing a Siamese networks inspired by the LeNet5 networks, and comparing the so-obtained features with the Euclidean distance. Since then, Siamese networks have been very popular in the literature. Several variants have been proposed, differing by their weight-sharing strategy [8] (Siamese vs Pseudo Siamese), combinations of the inputs [8, 9] (two channels input images vs multi-scale images), similarity functions (conventional distance [8, 10, 9, 11] or using metric layers [12]).

Another important aspect when training deep networks for patch matching is the objective function. It can be (a) the cross entropy (binary classification loss) [12] (b) the hinge loss [8] (c) the triplet loss [13, 14] to incorporate the notion of relative distance, relative distance [9] (d) the global loss which models the loss as two distributions (positive and negative) to be pushed away from each other [14].

More specifically, the question of multimodal patch matching has been investigated recently by several authors. [15, 16] suggested to concatenate the different modalities as different channels of the input data. [17] experimented the use of Siamese networks for the matching of visible/SAR patches. [18] studied the quadratic network, a variant of the Siamese network that takes 4 patches as input. In the context of cross-spectral face recognition [19] proposed two components (one before and another one after the feature extraction network) to allow the system to transform the NIR images into the VIS spectrum. As we write, Siamese networks are still seen as a reference for multimodal patch matching.

One important aspect of Siamese architectures is that the weights of feature extraction towers are shared between the inputs. This is to say that the network is trained to extract characteristics present in both modalities. In case of the Pseudo-Siamese architectures, the feature towers are not shared: contrarily to the Siamese networks, the motivation is to extract modality specific information in order to better discriminate the pair of inputs. Our motivation in this paper is to take advantage of these two complementary aspects and

---

propose a novel architecture, dubbed as TS-Net. It consists of two sub-networks, one Siamese and one Pseudo-Siamese networks. Their outputs are combined with a fully connected layer, acting as a weighting scheme between the modality specific information and the common information present in the input patches. The overall architecture is given Figure 1.

Our second contribution is to show that adding a constraint on the feature embedding, by means of a contrastive loss in the feature extraction tower, helps to boost the performance further. The idea is to encourage the network to bring projections of positive pairs closer in the Euclidean space. In the extreme case, this is equivalent to make two clusters of input pairs at the metric layers, allowing to easily separate them with an hyperplane instead of having to learn an arbitrarily complex boundary.

The rest of the paper is organized as follows: Section 2 introduces the network architecture and the training methodology. Section 3 discusses the datasets and presents the experimental validation of the approach. Finally, Section 4 concludes the paper.

2. THE PROPOSED THREE-STREAM NETWORK

As explained before, the proposed architecture for multimodal patch matching, denoted as the TS-Net architecture, is intended to combine the advantages of both Siamese and Pseudo-Siamese networks. The overall architecture of TS-Net is given Fig. 1(c). Each sub-network has 2 main parts: two feature extraction towers and a metric learning module. In the case of the Siamese network, the parameters of the feature extraction towers are shared, while for Pseudo-Siamese networks they are distinct. TS-Net takes a pair of patches as input, one from each modality, and predicts independently in each sub-network whether they are similar or not. Finally, the outputs of each sub-network are combined by an additional fully connected (FC) layer to produce the final prediction. In the next paragraphs, the different components of TS-Net are described and commented.

Feature extraction network. Each tower is based on convolutional and pooling layers to hierarchically extract high-level information from the input patches. We use max-pooling layers to reduce the dimensions of the feature maps by a factor of 2. At the end of the tower, we use a bottleneck (fully connected) layer to produce a compact output vector with 128 dimensions. Inspired from [12], we use $Relu$ activation as a non-linear activation function.

Tower Fusion. We observed in our experiments that subtracting the layers produced better performance than concatenating them, as in the original MatchNet. So the output of the feature extraction tower are element-wise subtracted before they are fed to the metric network.

Metric network. The metric learning part of the network

consists of three fully connected layers. It takes a vector of 128 dimensions and produces a vector of dimension two, suitable for binary classification.

Losses. We treat patch matching as a binary classification problem, as we observed it performs better (also observed by [9]) than learning a similarity function. Therefore, Siamese and Pseudo-Siamese parts of TS-Net are trained with binary cross-entropy loss functions.

One contribution of this paper is to introduce additional constraints, at the feature level, by means of a contrastive loss [20] enforcing the features coming from the two feature towers to be close to each other if the pair is positive. This will enable the features of positive pairs to be in the hypersphere and the features of the negative pairs to be outside the hypersphere.

The fusion of Siamese and Pseudo-Siamese networks is done by introducing an additional cross-entropy loss on the top of the two.

More formally, let (x_1, x_2) be the input pair of patches and y the class label. $y = 1$ means the pair is positive (similar patches), $y = 0$ means the pair is negative (different patches). We denote by L_{en} and L_{con} the cross-entropy and the contrastive loss, with:
(a) $L_{en} = y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})$ where $\hat{y}$ is output of the *Softmax* layer, and
(b) $L_{con} = y \frac{2}{Q} D^2 + (1 - y) 2Q e^{\frac{-2.77}{Q} D}$ where D is the Euclidean distance between features. **Q** is the margin to be optimized. The overall loss function is then given by:
$L = L_{tsnet_{en}} + L_{siam_{en}} + L_{pseudo_{en}} + \lambda L_{siam_{con}} + \beta L_{pseudo_{con}}$, with λ and β two cross-validated parameters in $[0, 1]$.

In multimodal settings, it is not always guaranteed that the two modalities can be projected into the same subspace. In practice, we observed that optimal performance is obtained for λ and β set to 10^{-2} (values obtained by cross validating the parameters on the validation set).

Implementation details. We initialize the weights of each convolutional layer using the Xavier initialization and all the FC layers with a truncated normal distribution ($stddev = 0.005$ and $mean = 0$, $bias = 0.1$). While the original MatchNet is trained with plain stochastic gradient descent, we found that training with 0.95 momentum produce equal or better performance. We train the network with $lr = 10^{-3}$ with L2 regularization of 10^{-3} with neither dropout nor $batchnorm$. Q is optimized experimentally on VeDAI validation set and set to be 50 for the other two datasets. During training, we observe that the λ and β parameters should be carefully set and the best performance we obtain is for $\lambda = 10^{-2}, \beta = 10^{-4}$ or $\beta = 10^{-2}$ on CUHK and NIR Scene (cross validation experiments). We use batch size of 32 and train with at least 150 epochs. All the experimentations are done using Tensorflow 1.4 with NVIDIA P100 or K80 GPU. Patches are normalized
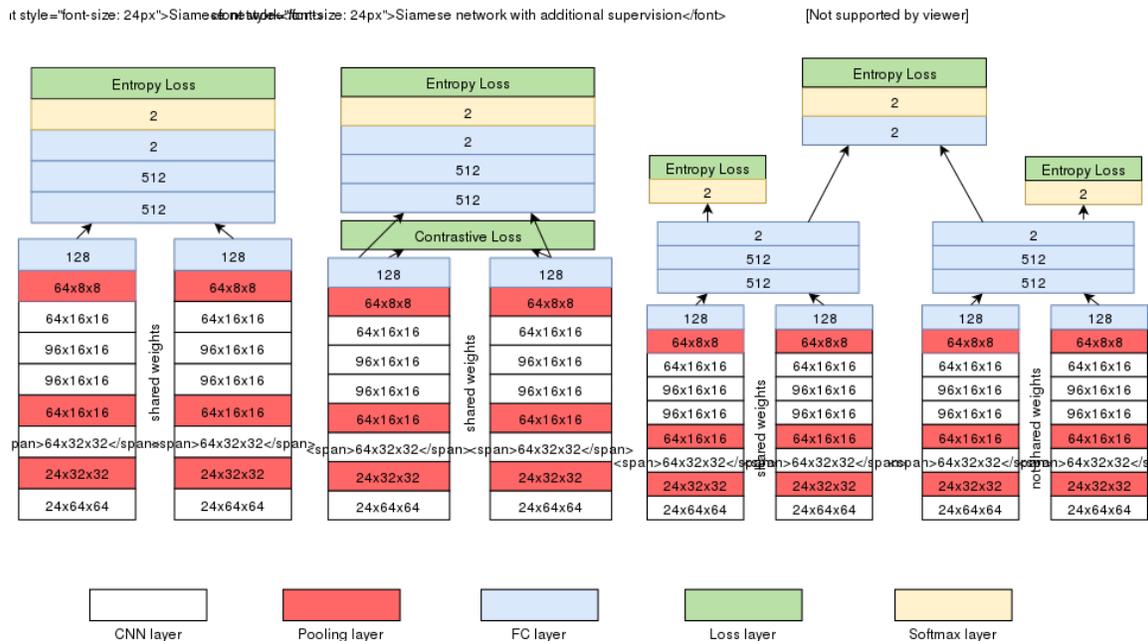
Fig. 1. The detailed architectures of (a) standard Siamese networks (b) Siamese networks with the proposed additional loss on the feature towers (c) the proposed TS-Net network with additional losses on the feature extraction tower and on the metric network. The numbers on each rectangle indicate the output size of this layer.

to have zero mean and unit standard deviation for each modality.

3. EXPERIMENTATIONS

Our aim in this section is to provide insights about TS-Net, its behavior and, more importantly, to draw a comparison with Siamese and Pseudo-Siamese networks, which are considered as a reference to this task. First, we run a series of experiments on the VeDAI dataset to validate TS-Net. It consists in evaluating different ways to fuse information either in the metric or after the feature extraction network. Next, we show that the gain in performance is not due to an increase of the number of parameters. Finally, we run experiments on three public datasets to experimentally validate our network and compare it to Siamese and Pseudo-Siamese networks. To report the performance, we employ the standard evaluation protocol defined in [6], namely the *95% error rate* criteria, abbreviated 95%ErrRate, which is the percentage of false matches present when 95% of all correct matches are detected. For each experimentation, we report the average performance with its standard deviation on at least 3 runs (Table 2) and 8 runs (Table 3).

Datasets The proposed approach is experimentally validated on three different datasets: VeDAI *, RGB-NIR Scene † and

CUHK ‡. These 3 datasets contain images from two different modalities. It is worth mentioning that these 3 datasets have been created for different tasks. Therefore, it will provide an opportunity to test and compare performance on a variety of fields. For instance, VeDAI is generally used for Vehicle Detection in Aerial Imagery while CUHK for face sketch synthesis/recognition. VeDAI, CUHK and RGB-NIR Scene contain respectively a total of 1246, 188 and 477 pairs of images.

Pairs of Patch Generation. For each dataset, the images are given as sets of aligned pairs (one image from each modality). To extract patches and form pairs, we uniformly sample each image using grid-like layout where each cell has a width and height of 64×64 pixels. This gives us a collection of corresponding positive patches. We randomly choose patches coming from different images to form negative pairs.

To make our patch matching experiments more realistic and challenging, we artificially augment our datasets by introducing some random affine transformations between the images of the same pair. For each pair, we generate three additional pairs using a random combination of: (i) Rotation (-12 to 12 degrees), (ii) Translation (-5 to 5 pixels on both axes) and (iii) Scale (0.8 to 0.99). For the validation and test set, we keep only one pair among the four, chosen randomly. Table 1 summarizes the number of train, test and validation

---

*https://downloads.greyc.fr/vedai/
†https://ivrl.epfl.ch/supplementary_material/

cvpr11/
‡http://mmlab.ie.cuhk.edu.hk/archive/facesketch.html

Table 1. Number of pairs of patches in the train, test and validation set, for each dataset. Each set contains 50% of positive pairs and 50% of negative ones.

Dataset	Train (70%)	Test (20%)	Validation (10%)
VeDAI	448k	128k	64k
CUHK	113k	32k	16k
NIR Scene	427k	122k	61k

Table 2. 95%ErrRate on VeDAI validation set using TS-Net. Rows: tower fusion after the feature extraction network (bottleneck layer), FC1, FC2 or FC3 of the metric layer. '1 Entropy' means there is only one classification loss at the top of the network. '3 Entropy': each sub-network also has his own classification loss. S*: Matchnet Network with the same number of parameters as TS-Net.

	3 Entropy losses	1 Entropy loss
FC3 (TS-Net)	**0.52 ± 0.07**	0.93 ± 0.05
FC2	0.62 ± 0.13	0.92 ± 0.05
FC1	0.74 ± 0.07	1.03 ± 0.06
Feature tower	n/a	1.05 ± 0.07
S*	n/a	1.01 ± 0.11

pairs of patches. Half are positive, half are negative.

Combining Siamese and Pseudo-Siamese networks. Our motivation is to find an efficient way to combine the information coming from the two sub-networks. We consider four options depending on whether this fusion (element-wise subtraction) is done (a) after the feature extraction tower (b) after the first (c) second or (d) third layer of the metric network. In the case of early fusion, all the following layers are kept as in MatchNet. Table 2 shows the performance given by each alternative. It also compares the performance obtained when 1 unique entropy loss ($tsnet_{en}$) is used, on the top of the network, with the performance obtained when each sub-network has, in addition, its own loss ($L_{tsnet_{en}} + L_{siam_{en}} + L_{pseudo_{en}}$). Based on these results, it is clear that the additional losses are important. The two additional losses help to guarantee the Siamese and the Pseudo-Siamese network learn complementary representation of the modalities. Consequently, this is the reason why having a late fusion (after FC3) is more beneficial. In addition, to guarantee that the gain in performance of TS-Net is not due to a larger number of parameters, we also provide the performance of MatchNet (noted as S* in Table 2) when we increase the number of parameters in the feature tower by a factor of 1.45 and the bottleneck by 2 to have exactly the same number of parameters as in TS-Net. Experimental results suggest that this is roughly equivalent to the performance of TS-Net without additional losses with fusion at the FC1 layer.

Influence of the contrastive loss. Table 3 presents the experimental results given by the three architectures: Siamese, Pseudo-Siamese and TS-Net network with/without the additional contrastive loss. In general, we observed that the error

Table 3. 95%ErrRate on the 3 datasets, for Siamese network alone (S), Pseudo-Siamese network alone (PS), TS-NET, without/with the additional contrastive loss (C).

Dataset	Vedai	CUHK	NIR Scene
S	1.16 ± 0.07	5.07 ± 0.46	14.35 ± 0.20
PS	1.50 ± 0.08	5.56 ± 0.36	16.05 ± 0.30
TS-Net	0.52 ± 0.07	3.58 ± 0.14	12.40 ± 0.34
S+C	0.84 ± 0.05	3.38 ± 0.20	13.17 ± 0.86
PS+C	1.37 ± 0.08	3.70 ± 0.14	15.60 ± 0.28
TS-Net+C	**0.45 ± 0.05**	**2.77 ± 0.07**	**11.86 ± 0.27**

can be reduced by up to 30 % by adding this loss, for any architecture and dataset. More importantly, this gain can be obtained with negligible computing costs and with little effort. During training, we found that the margin Q and the weighting value λ and β are crucial and need to be carefully cross-validated. We also consider replacing it by the classical contrastive loss. However it turned out to be very sensitive to gradient explosion. In addition, to make these parameters less sensitive during training, we tried to normalize the features before feeding into the loss function in order to maintain a fixed range of distances. Unfortunately, we observed some (marginal) drop in performance.

Comparison to Siamese and Pseudo-Siamese network. Intuitively, the Pseudo-Siamese network has more parameters and degree of freedom to project the two modalities onto the new subspace. Hence, it should produce better results compared to the Siamese network (See Table 3). However, in practice, we observed the opposite. We perform a grid search on the different parameters, regularization techniques (dropout, L2/L1), different losses (entropy/contrastive loss) with different strategy of combining the two towers (concatenation/subtraction). In all the experimentations, the Siamese network always outperform the Pseudo-Siamese network. This behavior has also been observed by [8, 15, 17]. When combining the Siamese and Pseudo-Siamese network, we notice significant improvement over the 3 datasets. On VeDAI and CUHK, the error is reduced by almost 50% not counting the additional loss at the feature level. On the three datasets, our approach outperforms the Siamese and Pseudo-Siamese networks. This fully justifies the competitiveness of our approach.

4. CONCLUSIONS

We proposed a novel architecture, called TS-Net, for multimodal patch matching. TS-Net consists of two sub-networks: a Siamese and Pseudo-Siamese network. Each of them is responsible for learning different types of complementary characteristics from both modalities. In addition, we showed that an additional loss, at the intermediate feature level, is beneficial at the price of only a small additional computational costs. Experimental results demonstrate the superiority of our

approach over Siamese and Pseudo-Siamese networks.

5. REFERENCES

[1] David G Lowe, "Object recognition from local scale-invariant features," in *ICCV*, 1999.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, "Surf: Speeded up robust features," in *ECCV*, 2006.

[3] Engin Tola, Vincent Lepetit, and Pascal Fua, "Daisy: An efficient dense descriptor applied to wide-baseline stereo," *IEEE PAMI*, vol. 32, no. 5, pp. 815–830, 2010.

[4] Jared Heinly, Enrique Dunn, and Jan-Michael Frahm, "Comparative evaluation of binary features," in *ECCV*. 2012.

[5] Prateek Jain, Brian Kulis, Jason V Davis, and Inderjit S Dhillon, "Metric and kernel learning using a linear transformation," *JMLR*, vol. 13, no. Mar, pp. 519–547, 2012.

[6] Matthew Brown, Gang Hua, and Simon Winder, "Discriminative learning of local image descriptors," *IEEE PAMI*, vol. 33, no. 1, pp. 43–57, 2011.

[7] Michael Jahrer, Michael Grabner, and Horst Bischof, "Learned local descriptors for recognition and matching," in *Computer Vision Winter Workshop*, 2008, vol. 2.

[8] Sergey Zagoruyko and Nikos Komodakis, "Learning to compare image patches via convolutional neural networks," in *CVPR*, 2015.

[9] Bin Fan Yurun Tian and Fuchao Wu, "L2-net: Deep learning of discriminative patch descriptor in euclidean space," in *CVPR*, 2017.

[10] Jure Zbontar and Yann LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *JMLR*, vol. 17, no. 1-32, pp. 2, 2016.

[11] Hani Altwaijry, Eduard Trulls, James Hays, Pascal Fua, and Serge Belongie, "Learning to match aerial images with deep attentive architectures," in *CVPR*, 2016.

[12] Xufeng Han, Thomas Leung, Yangqing Jia, Rahul Sukthankar, and Alexander C Berg, "Matchnet: Unifying feature and metric learning for patch-based matching," in *CVPR*, 2015.

[13] Vassileios Balntas, Edward Johns, Lilian Tang, and Krystian Mikolajczyk, "Pn-net: conjoined triple deep network for learning local image descriptors," *arXiv preprint arXiv:1601.05030*, 2016.

[14] BG Kumar, Gustavo Carneiro, Ian Reid, et al., "Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions," in *CVPR*, 2016.

[15] Cristhian A Aguilera, Francisco J Aguilera, Angel D Sappa, Cristhian Aguilera, and Ricardo Toledo, "Learning cross-spectral similarity measures with deep convolutional neural networks," in *CVPR Workshops*, 2016.

[16] Patricia L Suárez, Angel D Sappa, and Boris X Vintimilla, "Cross-spectral image patch similarity using convolutional neural network," in *ECMSM*. IEEE, 2017, pp. 1–5.

[17] Nina Merkle, Wenjie Luo, Stefan Auer, Rupert Müller, and Raquel Urtasun, "Exploiting deep matching and sar data for the geo-localization accuracy improvement of optical satellite images," *Remote Sensing*, vol. 9, no. 6, pp. 586, 2017.

[18] Cristhian A Aguilera, Angel D Sappa, Cristhian Aguilera, and Ricardo Toledo, "Cross-spectral local descriptors via quadruplet network," *Sensors*, vol. 17, no. 4, pp. 873, 2017.

[19] José Lezama, Qiang Qiu, and Guillermo Sapiro, "Not afraid of the dark: Nir-vis face recognition via cross-spectral hallucination and low-rank embedding," in *CVPR*, 2017, pp. 6807–6816.

[20] Sumit Chopra, Raia Hadsell, and Yann LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *CVPR*, 2005.