Learning cross-spectral similarity measures with deep convolutional neural networks

Cristhian A. Aguilera[1,2], Francisco J. Aguilera[3], Angel D. Sappa[1,4]
Cristhian Aguilera[5], Ricardo Toledo[1,2]

[1]Computer Vision Center, Edifici O, Campus UAB, 08193, Bellaterra, Spain
[2]DCC, Universitat Autònoma de Barcelona, Campus UAB, Bellaterra, Spain
[3]DIE, Víctor Lamas 1290, University of Concepción, Concepción, Chile
[4]Escuela Superior Politécnica del Litoral, ESPOL, FIEC, Guayaquil, Ecuador
[5]DIEE, Collao 1202, University of Bío-Bío, Concepción, Chile
[1] {caguilera, asappa, rtoledo}@cvc.uab.es, [3]fraguilera@udec.cl, [5]cristhia@ubiobio.cl

Abstract

The simultaneous use of images from different spectra can be helpful to improve the performance of many computer vision tasks. The core idea behind the usage of cross-spectral approaches is to take advantage of the strengths of each spectral band providing a richer representation of a scene, which cannot be obtained with just images from one spectral band. In this work we tackle the cross-spectral image similarity problem by using Convolutional Neural Networks (CNNs). We explore three different CNN architectures to compare the similarity of cross-spectral image patches. Specifically, we train each network with images from the visible and the near-infrared spectrum, and then test the result with two public cross-spectral datasets. Experimental results show that CNN approaches outperform the current state-of-art on both cross-spectral datasets. Additionally, our experiments show that some CNN architectures are capable of generalizing between different cross-spectral domains.

1. Introduction

Generally, the performance of computer vision applications strongly depends on lighting conditions, decreasing when dealing with low-light or no-light scenarios. This is an inherent limitation of just using images from the visible spectrum that can be overcome using images from different spectral bands or modalities [11]. Specifically, we are interested in cross-spectral applications, i.e., applications that use images from two different spectra to tackle in a more efficient way classical computer vision problems. For example, in surveillance systems, the use of thermal images can help to distinguish between background and foreground objects with similar visual appearance, while images from the visible spectrum can be used to discriminate between objects with a similar temperature [19].

(a) VIS (b) NIR (c) VIS (d) NIR

Figure 1. Common cross-spectral image matching challenges: *i)* texture may be lost between image pairs ((a) and (b)); and *ii)* the direction of the intensity gradients can change between image pairs ((c) and (d)). Images samples from [21]. **This figure is best viewed in color**.

In this paper, we focus on the cross-spectral matching problem by teaching CNNs to measure how similar two given patches are. This is a challenging task (see Figure 1), mainly due to the non-linear relationships between the corresponding pixel's intensities. For example, pixel intensity variations in the LWIR[1] spectrum are related to changes in the temperature of the objects, while pixel intensity variations in the visible spectrum are more related to color and texture [1]. Because of these natural differences between images acquired from different spectra, conventional methods used to compare image patches in a mono-spectral setting such as SIFT [15], SURF [4] or KAZE [3] end up providing a limited matching performance[12].

[1]Through this work the following acronyms will be used, VIS: VISible; NIR: Near InfraRed; LWIR: Long Wavelength InfraRed

Based on recent success in deep convolutional neural networks to measure the similarity between image patches in the VIS domain (mono-spectral case) [25, 10, 22], we train three CNN architectures to compute similarity measures between cross-spectral image patches. We train each network architecture utilizing images from the VIS and the NIR spectrum and then test the results with two public cross-spectral datasets. We follow a similar study to [25] but applied to cross-spectral cases.

The main contributions of this work are the following:

- Evaluations of CNN-based architectures to measure the similarity between cross-spectral image pairs. To our knowledge this is the first time that CNNs are used to compute patch similarity measures between natural cross-spectral image pairs—natural images refers to non-medical images and images not captured from satellites (remote sensing).
- We show that a network trained in the VIS-NIR cross-spectral domain can also be used to match features in the VIS-LWIR cross-spectral domain.
- Our trained networks and evaluation software are released to the community for further evaluation [2].

The rest of the manuscript is organized as follows. Section 2 describes the most recent work on image patch similarity approaches based on CNNs (for the VIS case) and cross-spectral image patch similarity. In Section 3 we detail all the network architectures trained and evaluated through this work. Section 4 describes the training settings used in our networks. In Section 5 we present the experimental results on two different public cross-spectral image datasets. Finally, conclusions are presented in Section 6.

2. Related Work

2.1. Cross-spectral similarity

To a great extent, most of the previous work in cross-spectral patch similarity consists in modifying and/or adapting conventional feature descriptor approaches used in the VIS spectrum. On this line, [8] proposes to compute the SIFT descriptor between $[0, \pi)$ instead of $[0, 2\pi)$ to become invariant to the intensity gradient directions[3]. Although this approach reduces the discriminative power of SIFT, it works well on VIS-NIR cross-spectral image pairs. Aguilera et al. [1] describe cross-spectral image pairs using a local version of the MPEG-7 texture descriptor EHD [16]. This approach is motivated by the non-linear relationship between images from the VIS and the LWIR spectral bands, giving more value to the information at the edges than to the pixel intensity values. In a similar way, [2] and [18] describe cross-spectral image pairs using a variant of the EHD descriptor, that consists in using as an input the image responses to different Log-Gabor filters instead of pixel intensity values. More recently, Kim et al. [12] introduce a local feature descriptor based on the image frequency response and local self-similarity.

The aforementioned methods can be categorized as local feature descriptors. Dense matching strategies also have been proposed in the literature. In contrast to local approaches, dense approaches try to globally optimize an objective function in order to find correspondences to all the pixels in an image. Ce Liu et al. [14] use a dense version of the SIFT descriptor and an objective function similar to the one used in optical flow to dense match images with non-linear intensity variations. In a similar way, [21] proposes a dense matching strategy based on variational approaches. The algorithm can be seen as a two-step method, where the first step is to globally estimate large displacements between the pixels and then use a local matching strategy to estimate the residual errors.

2.2. CNN-based patch similarity

CNN-based approaches are becoming the dominant paradigm in almost every computer vision task. CNNs have shown outstanding results in various and diverse computer vision tasks such as stereo vision [26], image classification [23] and recently also in local patch similarity [25], outperforming conventional hand-made approaches.

CNNs can be used to compare image patches in more ways than one. Fitcher et al. [9] evaluate the local descriptor dataset from [17] using as feature descriptor the first layer filters of the well known AlexNet network [13]. The resulting descriptor outperforms hand-made descriptors such as SIFT in almost every category, even though AlexNet was not trained to that task. Zbontar et al. [26] propose a siamese network architecture to compare image patches to do small-baseline stereo. More recently, [25] analyzes different CNN-based architectures, specifically designed and trained to compute images patch similarity, in a variety of problems and datasets. The trained networks outperformed the work of [9] in every category, becoming the base for more recent works such as [22]. Although the results of [25] are state-of-art regarding patch matching performance, in terms of runtime it is slower than conventional hand-made approaches. The aforementioned problem is addressed in [10], where the authors propose a generalization of the siamese networks in order to speed up the matching process. They divide the network in two parts, firstly a description network and then a metric network. In this way, each patch descriptor is just computed once, instead of multiple times such as in [25]. Another alternative was pro-

[2] https://github.com/ngunsu/lcsis

[3] This algorithm is referred to as GISIFT in the experimental result section of the current work

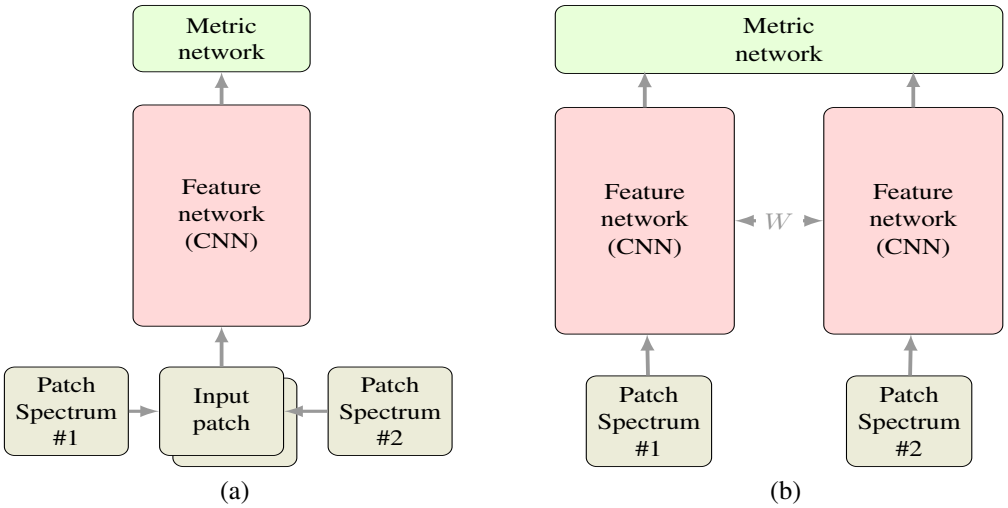


Figure 2. CNN architectures trained and evaluated during this work. (a) 2ch network, (b) siamese network if the weights W are shared and pseudo-siamese if the weights W are independent.

posed by [22], where the authors train a siamese network that compares the similarity of different image patches using the L2 distance. This makes the matching process much faster since it is possible to use fast approximate nearest neighbors algorithms to find the image correspondences and thus improve the overall matching runtime. However, it is not clear how well it performs when compared with CNNs that compute the distance between two image patches using a metric network. i.e., linear layers.

3. Network Architectures

In the current work three deep-network architectures are considered (i.e., 2-channel (2ch), siamese (siam) and pseudo-siamese (psiam), see Figure 1). Each one of these networks takes as input two image patches of size 64x64, where each patch belongs to a different spectra. The output is a scalar value that indicates the *distance* between the input patches.

We selected this three network architectures from [25] mainly for two reasons:

- Firstly, since the trained models are publicly available, we can compare our cross-spectral trained models with the same models trained for the VIS spectrum case. This will let us answer the following questions: Can CNNs trained in the VIS domain be used in cross-spectral applications without modifications? Are networks trained in the VIS domain similar to networks trained in the VIS-NIR domain?.

- Secondly, these network architectures can be easily adapted for the cross-spectral case, just setting each network input to a particular spectra—and keeping the same input relationship through the whole process, i.e., during training and testing.

Layer	Type	Output Dim	Kernel	Stride
1	convolution	96	7x7	1
2	ReLU	96	-	-
3	max-pooling	96	2x2	1
4	convolution	192	5x5	1
5	ReLU	192	-	-
6	max-pooling	192	2x2	1
7	convolution	256	3x3	1
8	ReLU	256	-	-
9	Linear	1	-	-

Table 1. 2ch network parameters.

3.1. 2-channel network

The 2ch network is depicted in Figure 1$(left)$. It takes as input an image patch of two channels, where each channel corresponds to one of the two patches to be compared. The network is composed of a series of convolution, ReLU and max-pooling layers, and a final linear layer that works as the metric network.

The 2ch network architecture combines the information of both cross-spectral image patches at the beginning of the network, processing in the next layers the combined information obtained in the first layer. Processing the data jointly from the first layer has proven to be the best solution in terms of feature matching performance in the visible domain (mono-spectral case) [25]. However, it is also one of the slowest solutions when matching local features. The 2ch parameters used in the current work are listed in Table 1.

(a) Mountain (b) Forest (c) Street (d) Indoor

Figure 3. Cross-spectral image pairs samples from the VIS-NIR dataset [6] used to create our cross-spectral patch dataset. The dataset is composed by 9 categories: country, field, forest, indoor, mountain, oldbuilding, street, urban and water. The first row corresponds to NIR images and the second row to VIS images. **This figure is best viewed in color**.

Layer	Type	Input Dim	Output Dim
9	Linear	512	512
10	Linear	512	1

Table 2. Siamese and pseudo-siamese metric network parameters.

3.2. Siamese network

Essentially, siamese networks are quite similar to traditional feature matching approaches, i.e., the network firstly computes feature descriptors for each patch and then evaluates the similarity between the descriptions using some trained metric. The siamese network consists of two CNNs feature networks with shared parameters that process each patch independently and a final metric network that acts as a distance metric (see Figure 1$(right)$). Each feature network is composed of a series of convolution, ReLU and max-pooling layers, while the metric network is composed of dense layers.

Siamese networks are slower than 2ch network at training, but can be faster at prediction. This is mainly due the fact that, once trained, it is possible to divide the network into two different stages and separately compute the feature description from the similarity measure.

In the current work the siamese feature networks have the same configuration than the 2ch network, just changing the metric network for the one described in Table 2.

3.3. Pseudo-siamese network

The pseudo-siamese network is essentially a siamese network but without shared parameters, i.e., each feature network is different from the other. This is important since the pseudo-siamese network can end up learning custom convolutional filters for each input spectrum, giving more flexibility to the network. The setting used in the current work are the same than the ones used in the siamese network.

4. Training

All the networks described in the previous section were trained in a supervised way. To that end, we built a cross-spectral image patch dataset[4] using the public VIS-NIR scene dataset from [6] (see Figure 3). The patches were extracted around interest points detected using SIFT in the VIS images. Half of the feature points were used as correct matches, extracting the same patch in the corresponding NIR image (note that both images are correctly registered), and the other half as false matches, extracting a random patch in the corresponding NIR image (see Figure 4). Table 3 shows the details of the patch dataset.

We use a margin criterion to train our networks as in [25]. The margin criterion optimizes the two-class classification hinge-based loss term described by the following equation:

$$\min_w \frac{\lambda}{2}||w||_2 + \sum_{i=1}^{N} max(0, 1 - y_i o_i^{net}), \qquad (1)$$

where w corresponds to the network weights, o_i^{net} is the network output for the i-th training sample, λ is the weight decay term and y is i-th training label. y can take two values, +1 if the i-th training sample is a correct match or -1 if it is a wrong one. In other words, we are training the networks in such a way that we expect large positive values at the output of the network when both patches are the same and small values when the patches are different.

[4] We provide a script to generate this dataset.

Category	# patches
country	277504
field	240896
forest	376832
indoor	60672
mountain	151296
oldbuilding	101376
street	164608
urban	147712
water	143104

Table 3. The VIS-NIR patch dataset used to train and evaluate our networks; it consists of more than 1 million of cross-spectral image pairs split up into 9 different categories. This table shows the number of cross-spectral patches per category.

(a) (b) (c) (d)

Figure 4. Image patches from the training set. The first row corresponds to grayscale images and the second row to NIR images. (a) and (b) are two samples of false matches and (c) and (d) two samples of correct matches.

Each model is trained using Stochastic Gradient Descent with a learning rate of 0.05, L2 weight decay (λ) of 0.0005, a momentum of 0.9 and batches of 256 samples. As recommended in [5], the training data was shuffled at the beginning of each epoch and each input patch is normalized by its intensity mean. All the patches from the *country* category were used to train the networks, were 80% of the data was used as training data and 20% of the data as validation. Additionally, we augmented the training data flipping the cross-spectral image pairs horizontally, vertically and rotating both images in 90 degrees—in order to increase the training data and prevent overfitting.

We implement all our code in Lua using the scientific computing framework Torch [7]. We trained the 2ch network during 2 days and the siamese and pseudo-siamese networks during 4 days each one on a 3.0 GHz Core I7 PC with a NVIDIA K40 GPU.

5. Experimental Results

Our trained networks were tested with two cross-spectral datasets—the VIS-NIR dataset described in Section 4 and the VIR-LWIR dataset from [2]. In all the experiments presented below the networks are referred to as 2ch-country (2-channel network model trained on the country sequence), siam-country (siamese network model trained on the country sequence) and psiam-country (pseudo-siamese network model trained on the country sequence). The country sequence was selected for training for two main reasons: *i)* in a preliminary evaluation stage the country sequence was one of the most difficult sequence in the VIS-NIR dataset; and *ii)* it is also one of the sequences with more data available. Additionally, for comparative evaluations with the state-of-art on VIS and cross-spectral patch similarity, we present results obtained with six of the trained networks presented in [25], SIFT [15], GISIFT[8], EHD[1] and LGHD[2]. These evaluations are intended for:

- Evaluating the performance of networks trained in the visible domain when they are used in a cross-spectral scenario—are the networks capable of generalizing?. The performance of these network architectures is additionally evaluated when they are trained with cross-spectral image pairs.

- Evaluating the generalization capability of networks. In other words, we would like to evaluate how well a network trained in a particular cross-spectral domain behaves in other cross-spectral domains. More specifically, we try to answer the following question: how well a VIS-NIR trained network performs on a VIS-LWIR dataset?

5.1. Local image patches (VIS-NIR)

We evaluate the performance of our networks using the false positive rate at 95% Recall (FPR95) on each category of the VIS-NIR scene dataset (as in [25]). To be fair, we do not include the country sequence in the mean FPR95 value provided at the right side of Table 4, since it was used to train our network.

The results of our tests are presented in Table 4. All our networks perform better than the ones trained just with images from the VIS spectrum. This is a not surprising result, since those networks were not trained for such a task, however it tell us that we cannot use this trained networks without modifications (*fine-tunning*) on cross-spectral applications. Moreover, the 2ch-country network outperforms all the other networks and descriptors in all the categories by a surprising margin. Clearly, as pointed out in [25], the key on the performance of the 2ch network is that the information is jointly processed right from the first layer.

Descriptor/Network	Country	Field	Forest	Indoor	Mountain	Oldbuilding	Street	Urban	Water	Mean
SIFT[15]	43.8	39.44	11.39	10.13	28.63	19.69	31.14	10.85	40.33	23.95
GISIFT[8]	31.44	34.75	16.63	10.63	19.52	12.54	21.8	7.21	25.78	18.60
EHD[1]	33.54	33.85	19.61	24.23	26.32	17.11	22.31	3.77	19.8	20.87
LGHD[2]	6.46	16.52	3.78	7.91	10.66	7.91	6.55	7.21	12.76	9.16
2ch liberty [25]	32.55	30.19	1.82	4.56	24.17	8.24	15.24	2.25	35.88	17.21
2ch notredame [25]	32.25	26.77	1.77	4.67	21.45	9.03	15.98	2.99	33.08	16.4
2ch yosemite [25]	38.97	36.37	1.76	5.54	30.75	12.63	17.21	4.35	38.64	20.69
siam liberty [25]	36.74	38.45	27.03	19.1	27.75	16.56	26.01	12.08	31.82	26.17
siam notredame [25]	34.84	36.28	25.65	13.37	24.42	16.77	25.2	11.65	30.03	24.24
siam yosemite [25]	30.15	33.79	20.86	20.87	22.21	18.77	21.58	17.27	27.75	23.69
2ch-country (ours)	**0.23**	**9.96**	**0.12**	**4.4**	**8.89**	**2.3**	**2.18**	**1.58**	**6.4**	**4.47**
siam-country (ours)	0.81	15.79	10.76	11.6	11.15	5.27	7.51	4.6	10.21	9.61
psiam-country (ours)	1.29	17.01	9.82	11.17	11.86	6.75	8.25	5.65	12.04	10.32

Table 4. Performance on the VIS-NIR local image patches dataset. The results correspond to the false positive rate at 95% Recall (FPR95). The smallest the better.

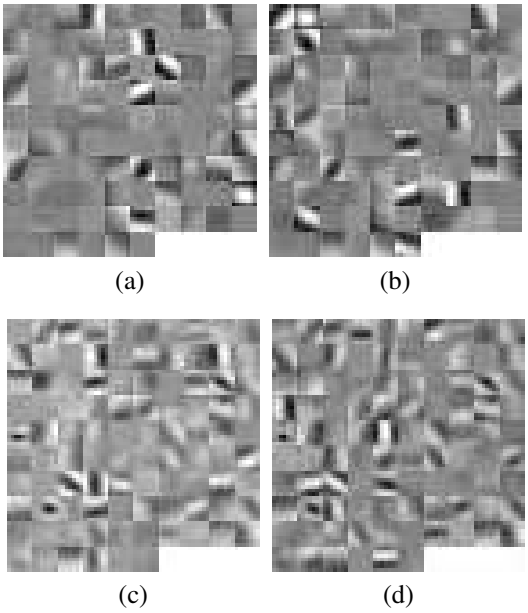


Figure 5. Visualization of the first layer filters of: (a) and (b) 2ch network filters trained in the visible spectrum domain (Yosemite); (c) and (d) 2ch network filters trained in the VIS-NIR cross-spectral domain (our best case).

A comparison of the first layer filters learned by the networks can be seen on Figure 5. Here we can see that our best model has learned similar filters to those presented in (a) and (b); somehow this means that the first layer features learned for image matching in different spectra are quite similar to those from grayscale image matching. We can also see from the filters that our trained network searches for lines and edges rather than textures, information that can be lost by switching to a different spectrum. This is interesting, since having similar first layer filters means that fine-tunning techniques can be applied to VIS similarity networks to work in cross-spectral domains. However, the success of these techniques will depend on how similar are the base datasets [24].

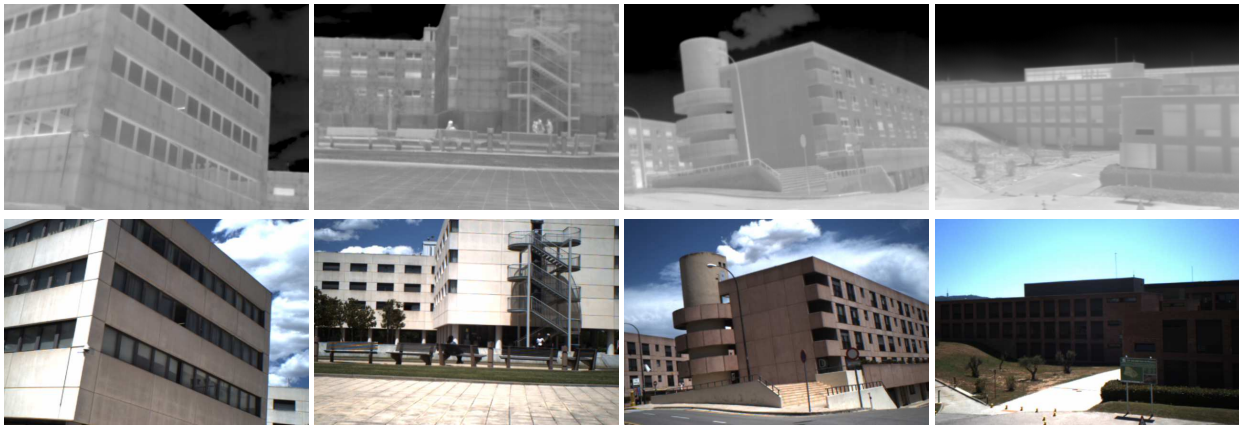
5.2. Local feature descriptors (VIS-LWIR)

We also evaluate our networks as replacement of local feature descriptors, i.e., we detect local feature points in each image pair, we extract patches of 64x64 around each feature point and then we do the matching using our trained networks in a brute-force manner. To that purpose, we selected the public VIS-LWIR cross-spectral dataset from [2] (see Figure 6), that consists of 44 VIS-LWIR registered image pairs taken around the campus of the Autonomous University of Barcelona. The resolution of the VIS and LWIR images is the same (639x431 pixels).

The selection of the local feature detector to be used in this evaluation was not an easy task. In general local features detected in images from different spectrum are different—a kind of low repeatability (see Figure 7). Hence, in order to minimize this inherent drawback of working with cross-spectral images, we end up using custom FAST [20] settings in each image pairs in order to have a similar response in both spectra[5]. This custom FAST settings increase the number of correct correspondences in the VIS-LWIR cross-spectral scenario; as already mentioned, cross-spectral feature point detection is still an open problem that needs special care in the tuning of user defined parameters.

We evaluate the performance of our networks using the mean average precision (mAP) as in the well-known local feature descriptor benchmark from [17], where the average

[5]We provide the keypoints and the groundtruth in the same webpage of the trained networks



(a) (b) (c) (d)

Figure 6. Cross-spectral image pairs samples from the VIS-LWIR dataset [2] used to evaluate our work. The dataset is composed by 44 VIS-LWIR image pairs taken around the campus of the Autonomous University of Barcelona. The first row corresponds to LWIR images and the second row to VIS images. **This figure is best viewed in color**.

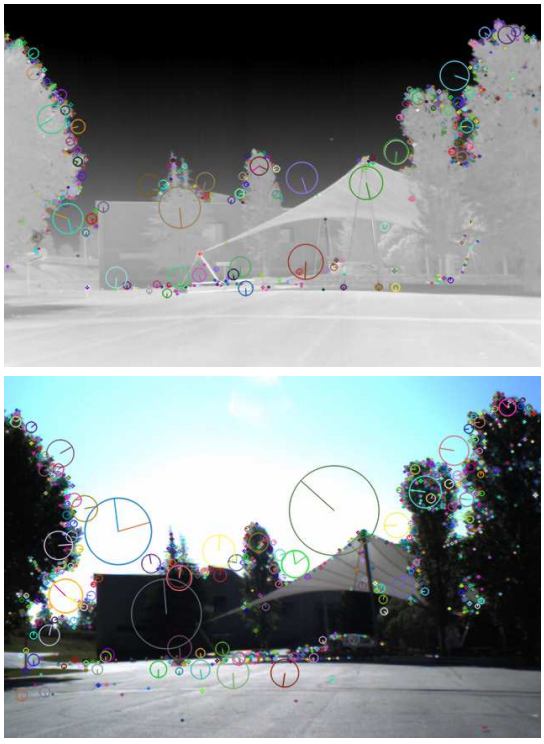


Figure 7. Visualization of cross-spectral feature detection using SIFT. The top is a LWIR image and the bottom the corresponding VIS image. **This figure is best viewed in color**.

precision corresponds to the area under the precision-recall curve—recall 1 correspond to the best possible result. Figure 8 shows the results of our evaluation. Similar to the VIS-NIR case the 2ch-country network outperformed all the other networks and cross-spectral descriptors. On the contrary, 2ch networks trained in the visible spectrum did not perform better than SIFT. Moreover, the results show that a 2ch network model trained in the VIS-NIR cross-spectral scenario can obtain a high mAP when matching image pairs from the VIS-LWIR domain. This generalization is mainly due to the fact that in both scenarios some of the same problems persist, like: loss of texture and differences in the gradient directions. The generalization capability is an important fact since the amount of images from the VIS-NIR spectra is considerably higher in comparison with those from the VIS-LWIR spectra.

6. Conclusions

Cross-spectral similarity measure is a challenging task. Our results show that using CNNs to determine the similarity between two patches from different spectra is feasible, and more important it outperforms other alternatives. As an interesting conclusion, in our experiments, a network trained on a VIS-NIR cross-spectral dataset has been later on used in a VIS-LWIR dataset, outperforming the state-of-art in cross-spectral image descriptors. This is an important results since the amount of public data available in the LWIR spectrum is smaller than in other spectra.

7. Acknowledgments

This work has been partially supported by the Spanish Government under Project TIN2014-56919-C3-2-R and the Chilean Government under project Fondef ID14I10364. Cristhian A. Aguilera has been supported by Universitat Autònoma de Barcelona.

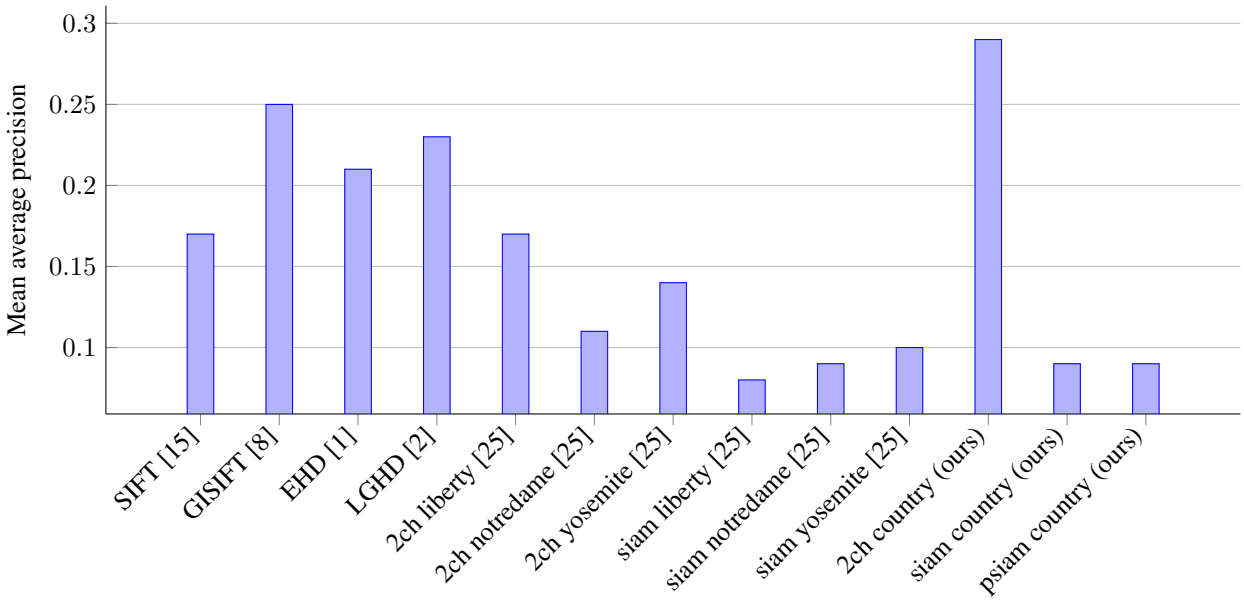


Figure 8. VIS-LWIR local feature descriptor performance. The results corresponds to the mean average precision over all the images in the dataset. The bigger the better.

References

[1] C. Aguilera, F. Barrera, F. Lumbreras, A. Sappa, and R. Toledo. Multispectral image feature points. *Sensors*, 12(9):12661–72, Jan. 2012. 1, 2, 5, 6

[2] C. A. Aguilera, A. D. Sappa, and R. Toledo. Lghd: A feature descriptor for matching across non-linear intensity variations. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 178–181, Sept 2015. 2, 5, 6, 7

[3] P. F. Alcantarilla, A. Bartoli, and A. J. Davison. Kaze features. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part VI*, ECCV'12, pages 214–227, Berlin, Heidelberg, 2012. Springer-Verlag. 1

[4] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008. 1

[5] L. Bottou. Stochastic gradient descent tricks. In *Neural Networks: Tricks of the Trade*, pages 421–436. Springer, 2012. 5

[6] M. Brown and S. Susstrunk. Multi-spectral sift for scene category recognition. In *CVPR*, pages 177–184, Colorado Springs, USA, Jun 2011. 4

[7] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, number EPFL-CONF-192376, 2011. 5

[8] D. Firmenichy, M. Brown, and S. Süsstrunk. Multispectral interest points for RGB-NIR image registration. In *ICIP*, pages 181–184, Brussels, Belgium, Sept. 2011. 2, 5, 6

[9] P. Fischer, A. Dosovitskiy, and T. Brox. Descriptor Matching with Convolutional Neural Networks: a Comparison to SIFT. *ArXiv e-prints*, May 2014. 2

[10] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg. Matchnet: Unifying feature and metric learning for patch-based matching. In *CVPR*, 2015. 2

[11] S. Kim, D. Min, B. Ham, S. Ryu, M. Do, and K. Sohn. Dasc: Dense adaptive self-correlation descriptor for multi-modal and multi-spectral correspondence. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 2103–2112, June 2015. 1

[12] S. Kim, S. Ryu, B. Ham, J. Kim, and K. Sohn. Local self-similarity frequency descriptor for multispectral feature matching. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 5746–5750, Oct 2014. 1, 2

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. 2

[14] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *PAMI*, 33(5):978–994, May 2011. 2

[15] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, Nov. 2004. 1, 5, 6

[16] B. Manjunath, J.-R. Ohm, V. Vasudevan, and A. Yamada. Color and texture descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):703–715, Jun 2001. 2

[17] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, Oct 2005. 2, 6

[18] T. Mouats, N. Aouf, A. Sappa, C. Aguilera, and R. Toledo. Multispectral stereo odometry. *ITS*, PP(99):1–15, Sep 2014. 2

[19] P. Pinggera, T. Breckon, and H. Bischof. On cross-spectral stereo matching using dense gradient features. In *Proc.*

British Machine Vision Conference, pages 526.1–526.12, September 2012. 1

[20] E. Rosten, R. Porter, and T. Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 32:105–119, 2010. 6

[21] X. Shen, L. Xu, Q. Zhang, and J. Jia. Multi-modal and Multi-spectral Registration for Natural Images. In *ECCV*, pages 309–324, Zurich, Switzerland, Sep 2014. 1, 2

[22] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer. Discriminative Learning of Deep Convolutional Feature Point Descriptors. In *Proceedings of the International Conference on Computer Vision (ICCV)*, Dec 2015. 2, 3

[23] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *CoRR*, abs/1409.4842, 2014. 2

[24] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3320–3328. Curran Associates, Inc., 2014. 6

[25] S. Zagoruyko and N. Komodakis. Learning to compare image patches via convolutional neural networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2, 3, 4, 5, 6

[26] J. Zbontar and Y. LeCun. Stereo matching by training a convolutional neural network to compare image patches. *arXiv preprint arXiv:1510.05970*, 2015. 2