Semi-supervised learning of Deep Metrics for Stereo Reconstruction

Applied Machine Learning Days
Jan 31st, 2017

Stepan Tulyakov

Anton Ivanov

François Fleuret

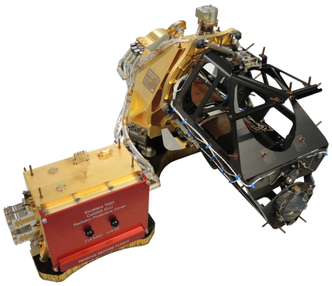
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE | eSPACE Space Engineering Center

idiap RESEARCH INSTITUTE

Introduction

Our motivation is to process images from the ExoMars Trace Gas Orbiter captured with the Colour and Stereo Surface Imaging System (CaSSIS).

ExoMars orbiter



PlanetS National Centre of Competence in Research

UNIVERSITÄT BERN

CaSSIS

State-of-the-art methods for stereo reconstruction use machine learning, but *we do not have labeled training data*.

Disclaimer: We did not process CaSSIS data *yet!*

Introduction

The standard machine-learning paradigm consists of using a training set

$$(x_n, y_n), \ n = 1, \dots, N$$

and to minimize a loss

$$L(w) = \sum_n l(f(x_n; w), y_n).$$

Can we do without the y_n ?

Yes, *if we can leverage prior knowledge about their joint structure.*

Introduction

If the y_n have a sequential and continuous structure, for instance through the maximum displacement

$$\max_n \|y_n - y_{n+1}\| \leq \Delta,$$

or the number of “switches”

$$\sum_n 1_{\{y_n \neq y_{n+1}\}} \leq S.$$

We can alternate:

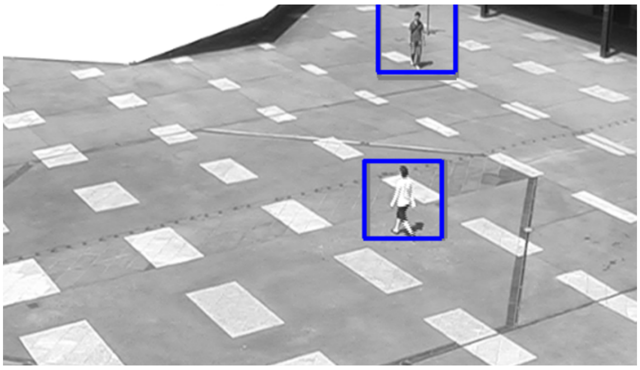
$$\begin{aligned} w^{u+1} &= \operatorname*{argmin}_w \sum_n l(f(x_n; w), y_n^u) \\ y^{u+1} &= \operatorname*{argmin}_{y, \mathcal{C}(y)} \sum_n l(f(x_n; w^{u+1}), y_n) \end{aligned}$$

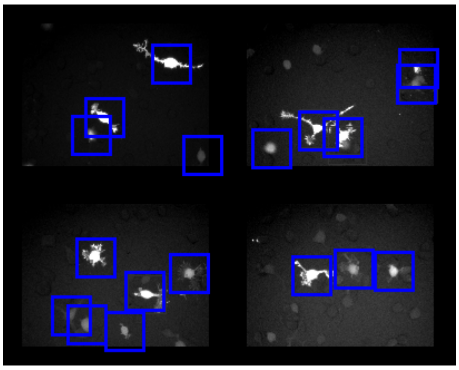
where $\mathcal{C}$ is the constraint over the admissible solutions.

Training a detector from video

Joint work with Karim Ali and David Hasler

We have access to a sparse labeling, and we know that trajectories are continuous.





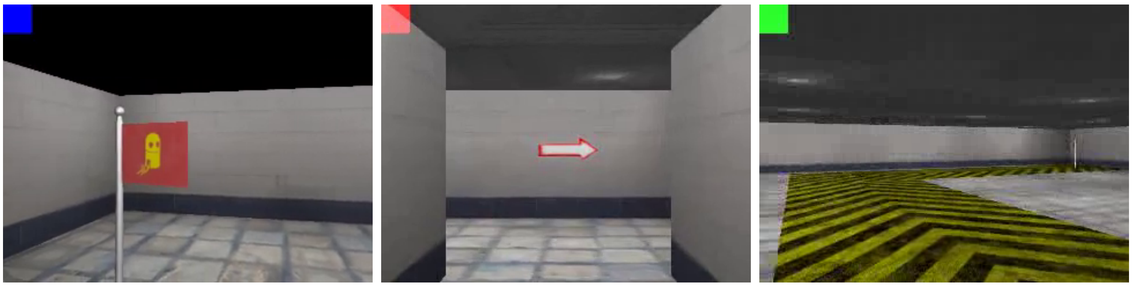
We minimize the same exponential loss in alternating a multi-target tracker and Boosting the predictor.

Performance are as good *or better* than using the full labeling.

Macro action discovery

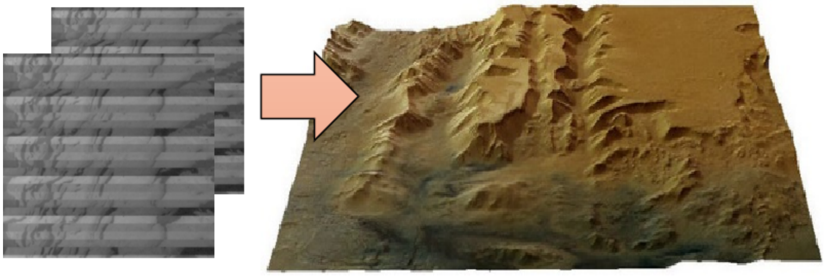
Joint work with Leonidas Lefakis

How to learn from a teacher when action choice depends on a hidden macro-state?



Learning a similarity metric for stereo reconstruction

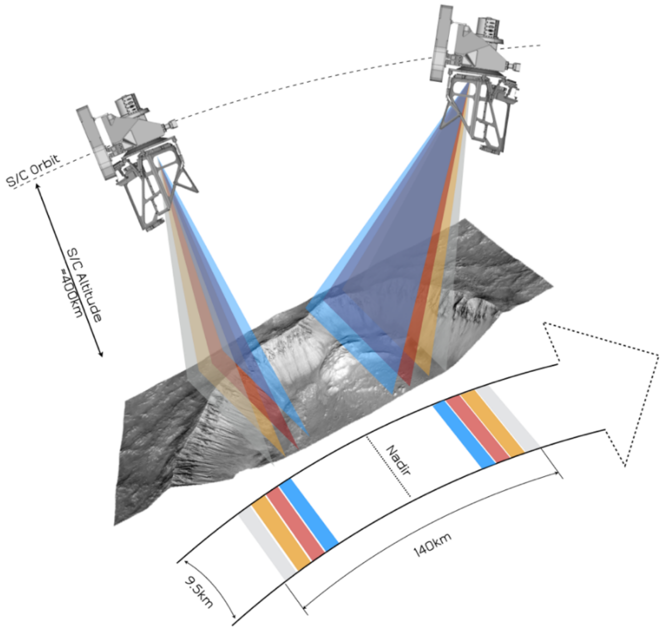
Stereo reconstruction consists of estimating the “depth” from two views taken from different angles.



At the core of the methods to do so lie similarity measures which can be learned from data.

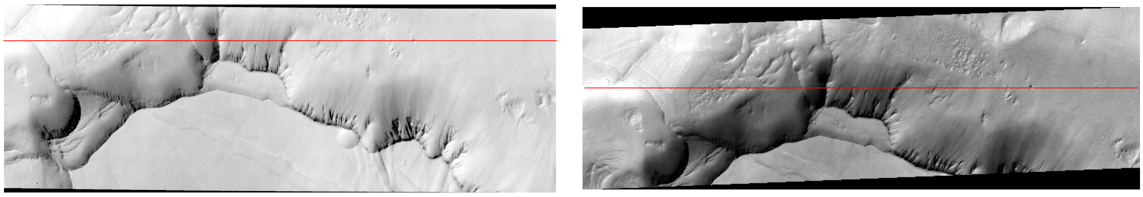
Learning a similarity metric for stereo reconstruction

CaSSIS moves physically in the orbiter so that points on the ground are seen from two different angles.



Learning a similarity metric for stereo reconstruction

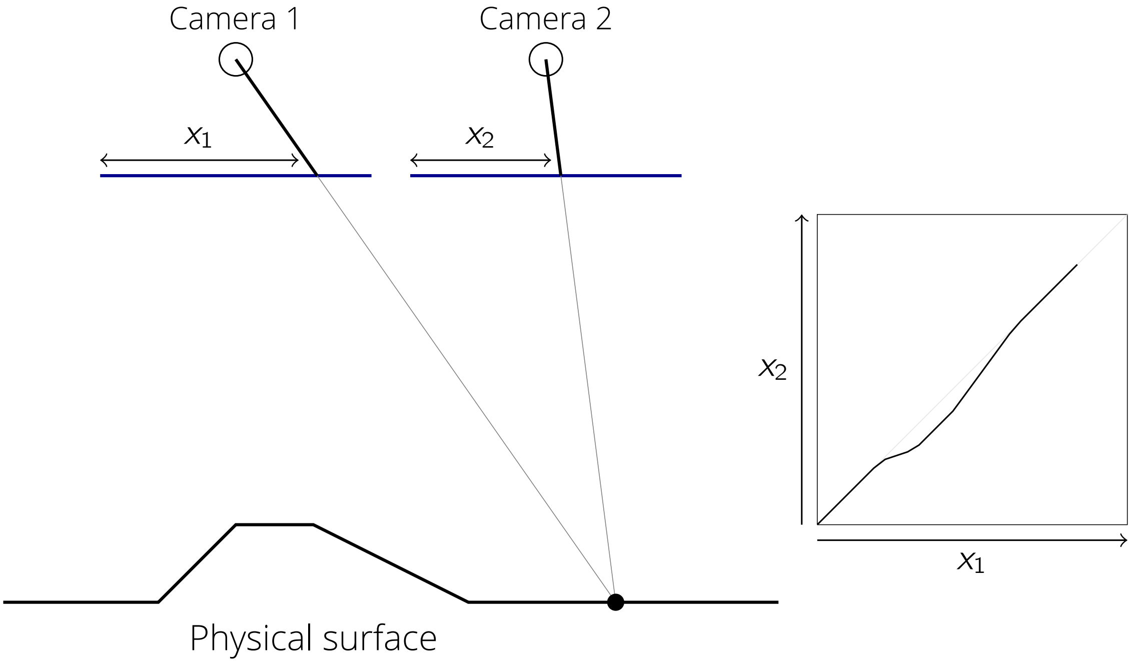
The optical constraints impose correspondences between images. They insure that to any point taken on an *epipolar line* in the first image, corresponds a point on an associated epipolar line in the second image.



Images can be rectified to make epipolar lines horizontal, as it is done here, but the actual correspondences depend on the surface height.

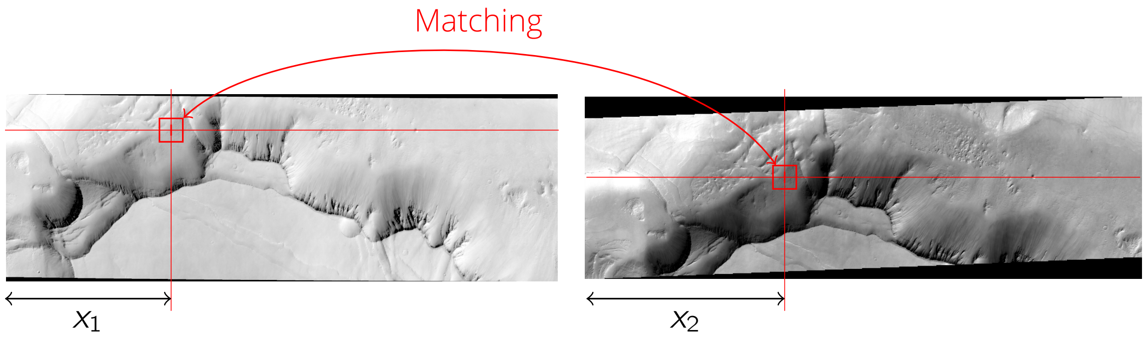
Learning a similarity metric for stereo reconstruction

The disparities in the correspondences reflect the terrain height.



Learning a similarity metric for stereo reconstruction

We want to match any point in one image to its physically corresponding one on the epipolar line in the other.



This can be achieved with a patch-matching cost:

$$\phi : \mathcal{P}^2 \to \mathbb{R}$$

where $\mathcal{P} = [0, 1]^{\Delta \times \Delta}$ is the set of gray-scale patches of size $\Delta \times \Delta$.

Learning a similarity metric for stereo reconstruction

A simple matching cost is the sum of absolute differences

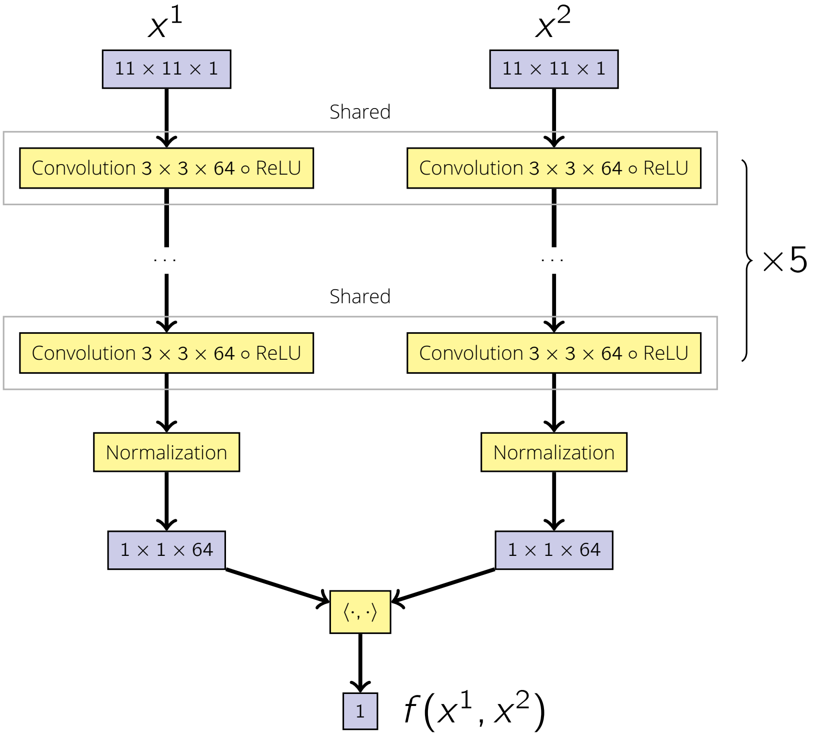
$$C_{SAD}(p_1, p_2) = \sum_{(i,j) \in \{1,\dots,\Delta\}^2} |p_1(i,j) - p_2(i,j)| \,,$$

that is the L^1 norm between patches.

Many hand-designed cost have been devised (cross-correlation, descriptors, etc.) but machine learning can exploit context more efficiently.

Learning a similarity metric for stereo reconstruction

We re-use Žbontar & Lecun's Siamese "fast network" (2016).


x^1
x^2
$11 \times 11 \times 1$
$11 \times 11 \times 1$
Shared
Convolution $3 \times 3 \times 64 \circ$ ReLU
Convolution $3 \times 3 \times 64 \circ$ ReLU
...
...
$\times 5$
Shared
Convolution $3 \times 3 \times 64 \circ$ ReLU
Convolution $3 \times 3 \times 64 \circ$ ReLU
Normalization
Normalization
$1 \times 1 \times 64$
$1 \times 1 \times 64$
$\langle \cdot, \cdot \rangle$
1
$f(x^1, x^2)$

Total of $\simeq 150k$ parameters.

Learning a similarity metric for stereo reconstruction

In the fully supervised case, disparities are measured with a laser scanner and for any patch x_n, the proper match x_n^+ is known.

Training relies on on a hinge loss, summed over triplets of samples

$$L(f) = \sum_n \max(0, 1 + f(x_n, x_n^-) - f(x_n, x_n^+))$$

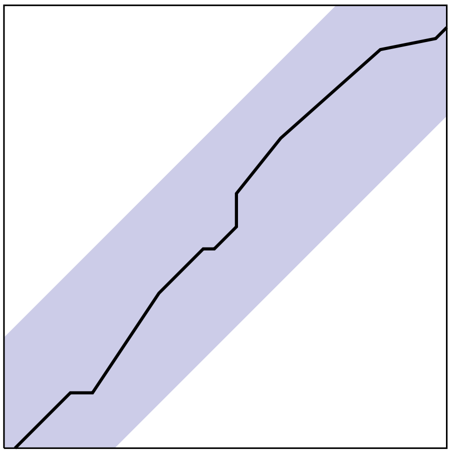
where x_n^- is an incorrect match.

Learning a similarity metric for stereo reconstruction

We propose to train *without ground-truth,* using the following constraints on the matches p^* between two corresponding epipolar lines:

- Upper bound on the disparity,
- continuity,
- ordering.

Given the patches from the two epipolar lines $(x^i_1, \dots, x^i_W) \in \mathcal{P}^W, \; i = 1, 2$ we compute the cost matrix $C(i, j) = f(x^1_i, x^2_j)$ and p^* under constraints:



Learning a similarity metric for stereo reconstruction

Our algorithm iterates the following steps:

1. Pick two corresponding epipolar lines at random $(x^i_1, \dots, x^i_W), i = 1, 2,$
2. compute the costs $C(1, 1), \dots, C(W, W)$,
3. compute the best constrained path $p^* \subset \{1, \dots, W\}^2$ with DP,
4. for each x_u retrieve
   - x^+_u its match in p^*,
   - x^-_u its best match outside p^*,
5. use back-propagation to minimize the summed hinge loss.

The network is initialized with the standard weight randomization.

Learning a similarity metric for stereo reconstruction

Our main measure of performance is the “winner take all” error on data-sets for which the ground truth is available.

Method	**KITTI’12**	**KITTI’15**	**MB**
MC-CNN fst	15.44%	15.38%	**29.94%**
MC-CNN-SS (ours)	**13.90%**	**14.08%**	30.06%

Learning a similarity metric for stereo reconstruction

Performance estimate through an external validating system with heavy post-processing.

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

MB Data-set

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

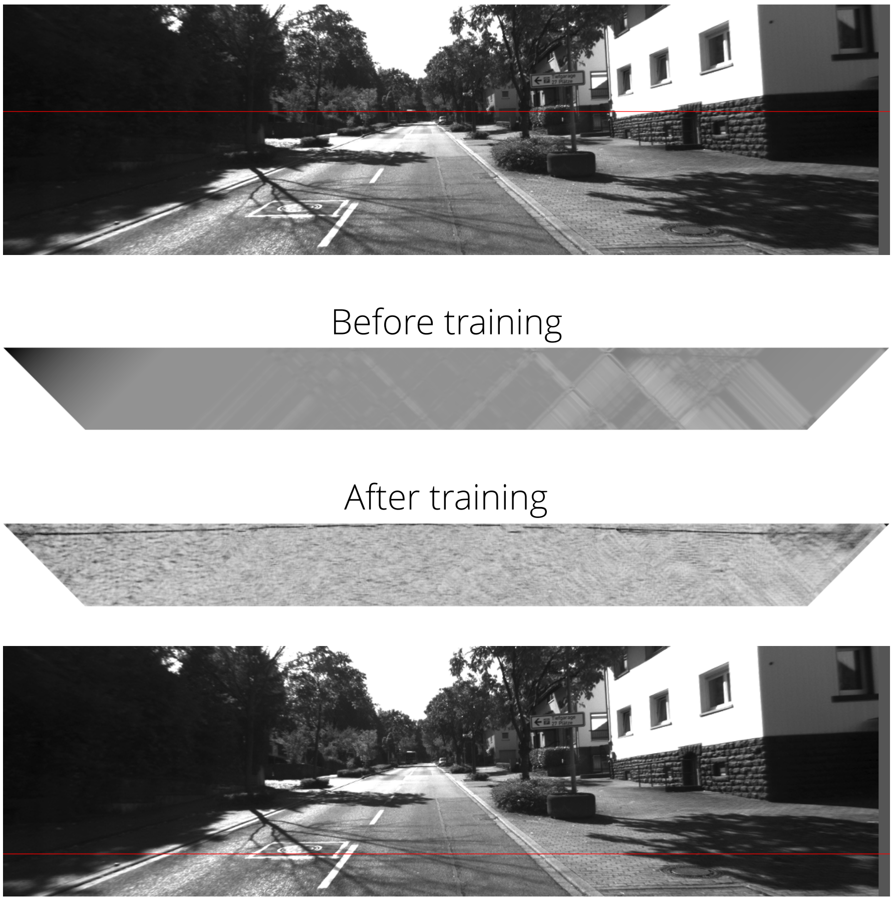
KITTI’12 Data-set

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

KITTI’15 Data-set

Learning a similarity metric for stereo reconstruction

The cost matrix get crispier after training.

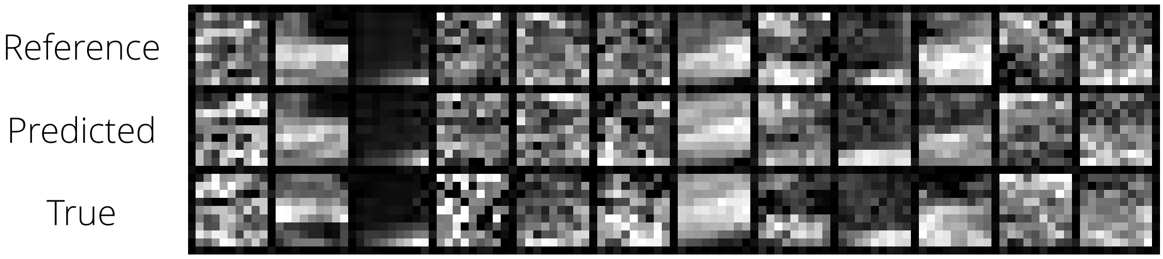


Before training

After training

Learning a similarity metric for stereo reconstruction

Mistakes are quite tricky to resolve based on patches only.



Conclusion

Without labeling, our approach allows to reach performance on par with the same model trained in a fully supervised manner, which opens the way for very large scale training.

The next step is to process the data from CaSSIS!

S. Tulyakov, A. Ivanov, and F. Fleuret. **Semi-supervised learning of deep metrics for stereo reconstruction.** *CoRR, abs/1612.00979,* 2016.

L. Lefakis and F. Fleuret. **Dynamic Programming Boosting for Discriminative Macro-Action Discovery.** *International Conference on Machine Learning (ICML),* pages 1548–1556, 2014.

K. Ali, D. Hasler, and F. Fleuret. **FlowBoost – Appearance Learning from Sparsely Annotated Video.** *IEEE international conference on Computer Vision and Pattern Recognition (CVPR),* pages 1433–1440, 2011.