# DeepSim-Nets: Deep Similarity Networks for Stereo Image Matching

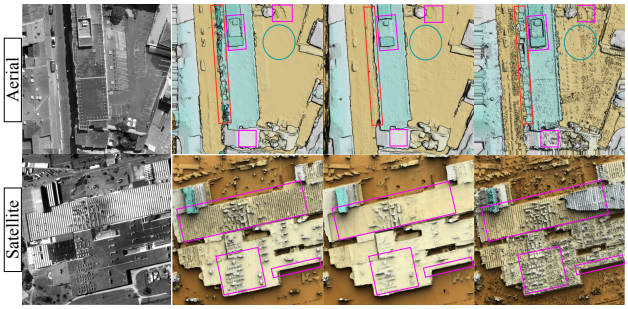
Mohamed Ali Chebbi[1,2] Ewelina Rupnik[2] Marc Pierrot-Deseilligny[2] Paul Lopes[1]

[1]Thales, France [2]Univ Gustave Eiffel, LASTIG, ENSG-IGN, F-94160 Saint-Mandé, France

https://dalichebbi.github.io/DeepSimNets/

## Abstract

*We present three multi-scale similarity learning architectures, or DeepSim networks. These models learn pixel-level matching with a contrastive loss and are agnostic to the geometry of the considered scene. We establish a middle ground between hybrid and end-to-end approaches by learning to densely allocate all corresponding pixels of an epipolar pair at once. Our features are learnt on large image tiles to be expressive and capture the scene's wider context. We also demonstrate that curated sample mining can enhance the overall robustness of the predicted similarities and improve the performance on radiometrically homogeneous areas. We run experiments on aerial and satellite datasets. Our DeepSim-Nets outperform the baseline hybrid approaches and generalize better to unseen scene geometries than end-to-end methods. Our flexible architecture can be readily adopted in standard multi-resolution image matching pipelines. The code is available at https://github.com/DaliCHEBBI/DeepSimNets.*




(a) Left Epipolar (b) **DeepSim-Net** (c) PSMNet (d) NCC

Figure 1. **Qualitative Results**. We show two disparity maps generated from *unseen* aerial (6cm) and satellite (WV-3, 30cm) stereo-pairs. On satellite data, our DeepSim-Net (b) performs best, while on aerial data the end-to-end PSMNet [3] (c) is best. The normalized cross-correlation (NCC) [15] (d) underperforms in both scenarios. On planar surfaces (◯) DeepSim-Net yields faithful reconstructions, whereas PSMNet adds residual artefacts. On aerial data, PSMNet learns to interpolate in occluded areas □, yet, it suppresses high-frequency details on satellite data and mis-constructs buildings' edges. Our DeepSim-Net recovers both buildings boundaries and fine details □.

## 1. Introduction

The availability of high quality large-scale stereo benchmark datasets [2, 14, 21] prompted many neural network architectures for stereo matching. These architectures can be classified into two categories: hybrid and *end-to-end*. To distinguish between matching and non-matching pixels, hybrid methods first extract features, then predict a similarity using a classifier (also referred to as *similarity learning*). To infer the optimal surface, the known semi-global matching (SGM) follows [11]. Hybrid methods show good generalization properties to unseen scenes. However, they operate on small patches which imposes convolutional neural networks (CNN) with limited expressivity.

End-to-end methods directly infer the surface from RGB images instead. They employ large image patches and deeper CNNs thus increase representations expressivity. Most importantly, to leverage geometry and context-aware disparities, end-to-end methods combine texture cues from 2D feature representations with shape cues captured within 3D CNNs. Their disadvantage is that they rely on positive and fixed disparity range cost volumes. In real world scenarios, disparities can take any values, depending on the geometry of the scene and the camera acquisition geometry.

In this paper, we revisit the self-supervised deep similarity learning approach. To address the fixed disparity range flaw of end-to-end methods, we decouple similarity learning from surface inference, thus our method is hybrid (see Fig. 2). To enhance the expressivity of our features we no longer consider the local neighborhood of a pixel (small patch) but use contextually richer epipolar image pairs as input, see Fig. 1. To our knowledge, the concept of deep similarity for stereo matching has not been introduced so far. Although counterintuitive, we show that an *off-the-shelf* segmentation network such as U-Net [13, 18] can be trained to learn similarity semantics, provided a proper sample mining scheme is adopted. Finally, to reduce the network size we propose a hard-coded multi-scale feature ex-

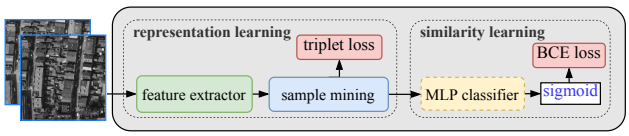



Figure 2. **DeepSim-Nets**. The feature extractor is one of the three backbone variants: U-Net 32, U-Net Attention, MS-AFF (see Fig. 3). Reference feature, positive feature and negative feature sets are generated by the sample mining block and serve both representation and similarity learning tasks. The above architecture is a building block of the multi-resolution pipeline in Fig. 6.

tractor (see Fig. 3) where specific non-weight sharing submodules learn specific scale cues. Unlike U-Net skip connection aggregation schemes, we employ an iterative attentional feature pooling mechanism to combine multi-scale features. Hence, we investigate the potential of implicit (U-Net) and explicit (ours) multi-scale learning. Note that multi-scale and multi-resolution are equivalent terms and we use them interchangeably throughout the paper.

To summarize, our main contributions are: (i) a new deep similarity learning architecture for stereo matching including a lightweight deep CNN architecture for feature learning that leverages hard-coded multi-scale features; (ii) a curated sample mining scheme to enable training deep architectures for our specified task; and (iii) a hybrid cooperative pipeline that benefits from the robustness of hand-crafted similarity measures for lower resolutions and rich semantics features for higher resolutions.

## 2. Related Works

**Similarity learning.** Similarity learning focuses on predicting pixels' resemblance and leaves the spatial aggregation on the cost volume to standard SGM [11, 15] or global optimization [19]. During training, matching and non-matching patches extracted from epipolar images are fed to a CNN in a self-supervised fashion [4,23] or in a fully supervised fashion [8, 22]. The task can be to either learn embeddings [4] or the matching metric [22, 23] or both [8]. In [4] the authors propose a two-scale CNN architecture that endows features with robustness leveraged at different scales. MC-CNN [23] is the baseline for stereo-matching contrastive learning and addresses both embedding and similarity learning. Match-Net [8] adds more context by using $64 \times 64$ patches albeit a single descriptor is extracted for the center pixel. Alternatively, multi-view patch features can be used to learn similarity for multi-view stereo [9]. Others perform random forest classification to fuse hand-crafted similarity filled cost volumes [1]. Note that because similarity learning is bound to small patches, thus has a restrained receptive field, it is more susceptible to matching ambiguities (i.e., henceforth referred to as *locality constraint*). To reduce stereo correspondence ambiguity, the similarities are backed by a regularization scheme that enforces surface regularity. An optimal regularization algorithm sets per-pixel edge-aware penalties which preserves thin structures and buildings outlines in the disparity map [11]. Among the major merits of this hybrid scheme is the similarity which remains unrelated to a specific matching geometry.

A different stream of work has been devoted to explicitly incorporating semantic information into the stereo matching task. Coupling 3D semantic segmentation with disparity estimation in multi-task learning can provide excellent results, especially on diachronic images [16]. However, coarse objects frontiers may lower the disparity map quality in a strong supervised setting.

**End-to-end disparity learning.** Among the first fully end-to-end stereo matching architectures is FlowNet [7]. To learn and predict the optical flow images are fed to a CNN either stacked on top of each other (e.g., FlowNetSimple) or considered independent and followed by a hard-coded correlation layer (e.g., FlowNetCorr). The more modern GC-Net [12], DeepPruner [6] and PSMNet [3] generate per-tile feature maps using a siamese CNN. Cost volume is subsequently built. GC-Net introduced a differentiable ArgMin (i.e., Soft-Argmin) that allows to train their network end-to-end. PSMNet builds upon that and introduces a deep 3D convolution Hourglass module to regularize the cost volume. The novelty of DeepPruner [6] is in exploiting the learnt representations to prune per-pixel disparity range, thus being able to serve real-time applications. Alternatively, SGM is revisited in GA-Nets [24] to leverage a differentiable optimization loss, and learn pixel-specific cost function parameters. Semi-global and local guided aggregation (SGA, LGA) layers are combined to balance regularity with edge awareness respectively.

So far, similarity learning being almost always governed by the locality constraint is not competitive with end-to-end methods that leverage both geometry and context [12].

## 3. Method

We introduced *DeepSim-Nets*, a family of neural network architectures that learn to predict similarity score maps between tiles of pixels. Fig. 2 highlights *DeepSim-Net*'s architecture consisting of a shared-backbone feature extractor, followed by a decision network that infers a similarity measure. The feature extractor consists of three variants: (1) U-Net 32; (2) U-Net *Attention* [13] performing gated attention feature pooling to aggregate encoder-decoder representations, and (3) our proposed explicit multi-scale feature learning module coupled with an adapted attentional pooling module [5] (see Fig. 3). Similarly to [23], we follow the self-supervised learning paradigm. Traditionally, to retrieve context-aware and discriminative similarity measures, stereo correspondences were bound to a local-context support windows. However, locality leads to similarity ambiguity, in particular on tex-

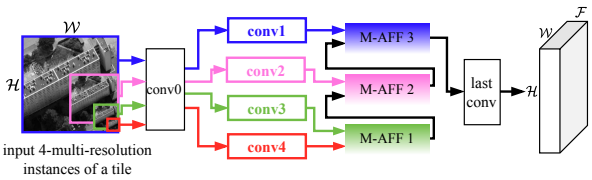



Figure 3. **Our Lightweight Feature Extractor**. Explicit **M**utli-**S**cale self-**A**ttentional **F**eature learning and **F**usion (MS-AFF). conv0 is a CNN with 3 3x3 convolutional blocks. **conv1**,**conv2**,**conv3** and **conv4** are composed of 3 3x3 convolutional residual blocks [10]. They do not share weights and handle 4 different resolution feature maps extracted by conv0. The resulting embeddings are iteratively fused from lower to higher resolutions using stacked attentional fusion blocks M-AFF[1-3] (see Fig. 4). The last_conv consists of 3 3x3 convolutional blocks to produce $\mathcal{H} \times \mathcal{W} \times \mathcal{F}$ features.

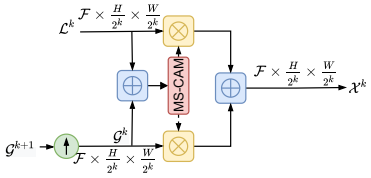



Figure 4. **Attentional Multi-scale Feature Fusion** (M-AFF). It is a building block of our lightweight feature extractor MS-AFF in Fig. 3. MS-CAM [5] learns to weight local and global embeddings contributions to the fused representations.$\oplus$ is the addition operator, $\otimes$ is the Hadamard multiplication operator.

tureless areas. To address this issue, we diverge from that idea. More specifically, non-local pixel embeddings that encapsulate similarity semantics are learnt by feeding large epipolar pairs $(768 \times 768)$ (i.e., *tiles*) to the proposed large receptive field feature backbone variants. The tiles are randomly cropped from the training stereo pairs.

A suitable sample mining scheme that not only considers a single pair of patches [4, 8, 23] but takes the whole set of features at once is then implemented. Hence, we allude to our sampling method as *ensembling* and outline it in Sec. 3.3. Finally, we address the similarity learning problem as a classification task where both matching and non-matching feature pairs are encouraged to be apart from each other. The same strategy is applied to the distributions of matching and non-matching similarity metrics learnt by the subsequent Multi-Layer-Perceptron (MLP) classifier.

### 3.1. Representation learning

The feature extractor takes a grayscale $\mathcal{H} \times \mathcal{W}$ image (i.e, tile) and outputs $\mathcal{H} \times \mathcal{W} \times \mathcal{F}$ feature map (see Fig. 2). We compute similarity scores along the epipolar line between a reference feature $f^l_{x,y} \in \mathbb{R}^{\mathcal{F}}$ from the left epipolar pair (i.e., left $(l)$ tile) and a set of possible features drawn from the right epipolar pair (i.e., right $(r)$ tile) $f^r_{x-i,y} \in \mathbb{R}^{\mathcal{F}}$ with $i \in [d_{min}, d_{max}]$ defining the disparity search space. $(x, y)$ are pixel locations. The normalized dot product $\langle ., . \rangle$ between a pair of features is inherently a similarity measure as it equals the cosine of the angle between them. Therefore, by training the backbone feature extractor to yield high similarity scores for matching feature pairs and low similarity scores for mismatching ones, the network learns robust and discriminative features that encapsulate similarity cues. It follows that for a set of reference features $\mathcal{X}$ drawn from the left tile, a set of matching features $\mathcal{X}_+$ and a set of non-matching features $\mathcal{X}_-$, all generated from the right tile, our triplet loss is:

$$\mathcal{L}_3 = \sum_{(i,j)\in\mathcal{X}} \mathcal{O}(\mathcal{S}^{i,j}_- - \mathcal{S}^{i,j}_+ + m, 0), \quad (1)$$

where $(i, j)$ denote feature coordinates in the reference feature map, $\mathcal{S}^{i,j}_- = \left\langle \mathcal{X}^{i,j}_-, \mathcal{X}^{i,j} \right\rangle$, $\mathcal{S}^{i,j}_+ = \left\langle \mathcal{X}^{i,j}_+, \mathcal{X}^{i,j} \right\rangle$ are cosine similarities between features; $m$ is the separation margin; and $\mathcal{O}$ is the element-wise max operator. We set $m$ empirically to 0.3 and keep it fixed for all experiments.

**Attentional feature pooling.** Combining multiple and complementary types of features to obtain semantically stronger representations has proven beneficial in many application domains [5, 13, 17]. Similarly, two sets of spatially consistent feature maps computed at different image scales encapsulate complementary cues as their respective fictive receptive fields differ. This observation was used in the U-Net architecture [18] where low and high level features are concatenated via long skip connections. Here, rather than blindly concatenating the multi-scale features as does U-Net , we introduce an aggregation strategy through a Multi-Scale self-Attention Feature Fusion (MS-AFF). Thanks to this explicit fusion we reduce the number of parameters by a factor of 10.

MS-AFF's key idea is presented in Fig. 3, where $\mathcal{L}^k$ is a feature map of shape $\frac{H}{2^k} \times \frac{W}{2^k} \times \mathcal{F}$ denoted as local feature map and $\mathcal{G}^{k+1}$ is a more global feature of shape $\frac{H}{2^{k+1}} \times \frac{W}{2^{k+1}} \times \mathcal{F}$. Based on the multi-scale attention feature fusion module (MS-CAM) [5] denoted by $\mathcal{M}$, we refine the feature map $\mathcal{X}^k$ at scale $k$ using the formula:

$$\mathcal{X}^k = \mathcal{M}(\mathcal{L}^k \oplus \mathcal{G}^k) \otimes \mathcal{L}^k + (1 - \mathcal{M}(\mathcal{L}^k \oplus \mathcal{G}^k)) \otimes \mathcal{G}^k, \quad (2)$$

where $\mathcal{G}^k$ is a one level up-scaled version of $\mathcal{G}^{k+1}$. By extending this fusion concept to pyramidal features maps, multiple attentional fusion blocks can be stacked on top of each other moving from coarser low-level contextually rich features to fine-grained high resolution features (see Fig. 4). At each step, the result of the last aggregation is considered as a global feature for the next fusion block.

### 3.2. Similarity learning

Our goal is to learn a powerful similarity function that predicts the matching likelihood between two embeddings. We believe that to describe complex relationships between corresponding pixels, the baseline dot product is insuffi-

cient. To that end, we feed the learnt representations to an MLP module acting as the decision function. Supervision is accomplished by the binary cross entropy (BCE) loss [23]. Following the previously introduced notation, given the triplet of feature sets $\mathcal{X}$,$\mathcal{X}_+$ and $\mathcal{X}_-$, we formulate the per-tile BCE loss $\mathcal{L}_{\mathcal{BCE}}$ as:

$$\mathcal{L}_{BCE} = -\sum_{(i,j)\in\mathcal{X}} \mathcal{Y}_-^{i,j} \log(1-\mathcal{S}_-^{i,j}) + \mathcal{Y}_+^{i,j} \log(\mathcal{S}_+^{i,j}), \quad (3)$$

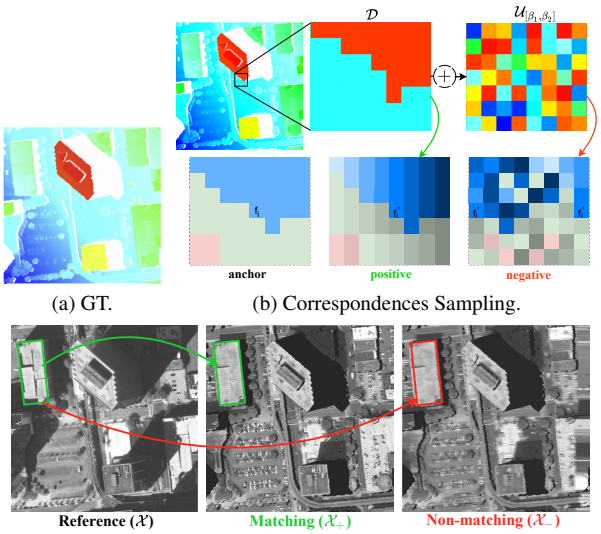
where $\mathcal{S}_-^{i,j} = \Phi(\mathcal{C}(\mathcal{X}_-^{i,j}, \mathcal{X}^{i,j}))$, $\mathcal{S}_+^{i,j} = \Phi(\mathcal{C}(\mathcal{X}_+^{i,j}, \mathcal{X}^{i,j}))$. $\mathcal{C}$ is the concatenation operator and $\Phi$ is the MLP that maps features from $\mathbb{R}^{\mathcal{F}\times 2}$ to $\mathbb{R}$; $\mathcal{Y}_+$ and $\mathcal{Y}_-$ are positive and negative sample definition masks, respectively. A sample definition mask is a binary mask that defines the matching features locations included in the loss computation.

### 3.3. Sample mining

**Ensembles approach.** Our sampling technique is designed to operate at a tile level, ensuring that the features learned by the network are consistent across entire objects (e.g., buildings, roads, etc.). Moreover, our approach involves presenting an *ensemble* of samples in a single gradient update, which not only adds more spatial context but also prevents overfitting. Specifically, in one gradient step a feature can appear as a positive match to one feature, and at the same time as a negative match to several other features (see Fig. 5). Patch-based shallow networks cannot capture spatial relationships in large objects, unlike our method.

**Sample Mining.** The quality and density of the dataset can impact the sample mining. For instance, coarse optical-LiDAR registration may produce false pixel correspondences, which can confuse positive and negative examples. To address this, we sample positive pixel examples around ground truth back-projected LiDAR points and negative pixel examples slightly further away from the ground truth. Additionally, we densify the LiDAR ground truth data using Delaunay interpolation to match the image data density, preserving high frequency changes and occlusion constraints while being a purely geometric approach. Because contrastive learning [20] is highly sensitive to sample mining, we carefully adjust the gap between positive and negative samples to prevent the selection of easy negatives throughout all training phases. We begin by extracting negative samples that are far from the positive ones and gradually tighten the classification difficulty by reducing the distance gap between positives and negatives. To sample positive and negative feature sets, we use the reference features from the left tile $\mathcal{X}$ and a disparity ground truth map $\mathcal{D}$ as follows (see Fig. 5):

$$\left\{ \begin{array}{l} \mathcal{X}_+ = \mathcal{X} \pm (\mathcal{D} + \mathcal{U}_{[-\alpha,\alpha]}) \\ \mathcal{X}_- = \mathcal{X} \pm (\mathcal{D} + \mathcal{U}_{[\beta_1,\beta_2]}) \\ \alpha < \beta_1 < \beta_2 \end{array} \right\}, \quad (4)$$




(a) GT. (b) Correspondences Sampling.

(c) Triplet Feature Sets Generation.

Figure 5. **Ensembles Sample Mining Toy Example**. (a) Ground truth (GT) disparities define matching pixels mappings ($\mathcal{D}$). (b) Let $(f_i, f_i^{'})$ be a feature pair labelled as positive. Negative feature pair is picked randomly along the epipolar line in the vicinity of $f_i^{'}$. Note that $f_i^{'}$ can have both *matching* (positive) and *non-matching* (negative) states in the **same** gradient update. This enforces the matching uniqueness constraint while preventing overfitting. (c) At tile level, the feature map is warped based on ground truth disparities ($\rightarrow$), and denoted as the matching feature map ($\mathcal{X}_+$). The non-matching feature map ($\mathcal{X}_-$) is obtained by warping the feature ($\rightarrow$) map using random disparity offsets ($\mathcal{D}+\mathcal{U}_{[\beta_1,\beta_2]}$).

where $\mathcal{U}_{[-\alpha,\alpha]}$ and $\mathcal{U}_{[\beta_1,\beta_2]}$ are uniform distributions of matching and non-matching sampled positions, respectively; $\alpha$ denotes a symmetric interval around the ground truth for positive samples whereas $\beta_1$ and $\beta_2$ are the negative sampling interval bounds.

**Occlusion handling.** As our main task is to learn dense similarities by means of binary classification, occlusions should be handled carefully since the correspondence problem is violated in these regions. In patch-based learning approaches, it is addressed by training exclusively on samples extracted in non-occluded areas [4, 8, 23], while traditional correlation-based dense matching yields occlusion masks by applying a hard threshold on the computed correlation values [15]. This approach is effective if similarities in occlusions remain low, which is true for NCC or census similarity metrics. However, for our deep-learnt similarity, resemblance is more than visual and is deduced from the entire feature map structure. Hence, the model should be explicitly told that some features do not have their corresponding matches. We account for occlusion by labelling sample features extracted from these regions as negatives. By reformulating our training losses, the network is not incentivized to match features in occluded areas which we accomplish by penalizing the inferred similarity measures

during training. Put differently, the network is trained to output low similarity scores in occlusion regions. Note that our aim is to filter out occluded areas through similarity and not to enhance the underlying surface regularity. Our new losses are reformulated as follows:

$$\begin{aligned}\mathcal{L}_{3_{All}} &= \mathcal{L}_{3_{nocc}} + \mathcal{L}_{3_{occ}} \\ &= \sum_{(i,j)\in\mathcal{X}_{nocc}} \mathcal{O}(\mathcal{S}_{-}^{i,j} - \mathcal{S}_{+}^{i,j} + m, 0) \\ &+ \sum_{(i,j)\in\mathcal{X}_{occ}} \mathcal{O}(\mathcal{S}_{1-}^{i,j} + \mathcal{S}_{2-}^{i,j}, 0)\ ,\end{aligned} \tag{5}$$

where $\mathcal{S}_{1-}^{i,j} = \left\langle \mathcal{X}_{1-}^{i,j}, \mathcal{X}_{occ}^{i,j} \right\rangle$ and $\mathcal{S}_{2-}^{i,j} = \left\langle \mathcal{X}_{2-}^{i,j}, \mathcal{X}_{occ}^{i,j} \right\rangle$ are cosine similarities between a reference feature $\mathcal{X}_{occ}^{i,j}$ located at occlusions and features $\mathcal{X}_{1-}^{i,j}$ and $\mathcal{X}_{2-}^{i,j}$ sampled from the right feature map. The BCE loss is expressed accordingly:

$$\begin{aligned}\mathcal{L}_{BCE_{All}} &= \mathcal{L}_{BCE_{nocc}} + \mathcal{L}_{BCE_{occ}} \\ &= -\sum_{(i,j)\in\mathcal{X}_{nocc}} \mathcal{Y}_{-}^{i,j} \log(1 - \mathcal{S}_{-}^{i,j}) + \mathcal{Y}_{+}^{i,j} \log(\mathcal{S}_{+}^{i,j}) \\ &-\sum_{(i,j)\in\mathcal{X}_{occ}} \mathcal{Y}_{-}^{i,j} \Big( \log(1 - \mathcal{S}_{1-}^{i,j}) + \log(1 - \mathcal{S}_{2-}^{i,j}) \Big)\ ,\end{aligned} \tag{6}$$

### 3.4. Learning strategy

We apply the same sampling scheme to both the feature backbone and the decision network training, alternating between the triplet and the BCE loss. We adopt a differential learning strategy to train the whole model. More specifically, we set the initial learning rate to *0.001* and progressively divide it by a factor of 10 for later (decoder), intermediate (bottleneck) and earlier (encoder) backbone parameters. Following a coarse to fine training scheme, we set the matching pixel locations sampling interval $\alpha$ to $\{1, 0\}$ and the non-matching pixel locations sampling intervals $\beta_1$ and $\beta_2$ progressively to $\{2, 8\}$, $\{2, 6\}$, $\{1, 5\}$ and $\{1, 4\}$, respectively. This gradual tightening scheme allows to leverage easy negatives at the beginning of training and helps the network learn fast. Then, we track harder negatives by reducing the distance to the ground truth locations. By doing so, we incite that features or learned similarities are not only distinctive far away from correct matches but also within their vicinity.

All training scenarios are run for 50 epochs per each sampling interval. Finally, we perform a last full tight configuration training (i.e., backbone, MLP, $\alpha = 0$, $\beta_1 = 1$, $\beta_2 = 4$). To avoid overfitting, we train our model on tile subsets that are randomly extracted in the course of training. This guarantees that the model sees a quasi-different sample of the dataset at each epoch. Note that all models have been initially trained on non-occluded masked areas. The occlusion sensitive loss functions (Eqs. (5) and (6)) were engaged in the very final training.

## 4. Experiments

**Implementation details.** DeepSim-Nets predict similarities which are used as input to a semi-global matching in the post-processing. The goal is to reduce the underlying noise and penalize disparity jumps within a local neighborhood of the cost structure. This regularization is performed using MicMac's SGM implementation [15]. To keep inference memory-friendly and fit for large scale production pipelines, our architecture is integrated into a multi-resolution iterative approach, also present in [15] (see Fig. 6). This approach involves exploring *n-scale* images drawn from the original full resolution image and generating multi-scale aggregated features through concatenation or self-attention mechanism (see Sec. 3.1). Our learning models are activated from scale 3. More explicitly, given a full resolution epipolar pair of dimensions $\mathcal{H} \times \mathcal{W}$, our models are deployed from resolution $\frac{\mathcal{H}}{4} \times \frac{\mathcal{W}}{4}$. We currently train DeepSim-Nets on a mix of 8-bit and 16-bit single-channel images because our focus is 3D reconstruction and high-resolution satellite sensors are by design panchromatic. However, the network can easily be extended to more channels.

**Datasets.** We perform training on the aerial dataset [21,25] consisting of *30,841*; *3164* and *607* pairs of epipolar images over Dublin, Enschede and Vaihingen, respectively. All tiles sizes are set to *1024* $\times$ *1024*. A ground truth reprojected LiDAR disparity map is given for each epipolar pair. Evaluation is performed on aerial stereo pairs (Ground Sampling Distance GSD=8cm) over Toulouse, satellite stereo pairs over Buenos Aires (WV-3, GSD=30 cm) and Montpellier (Pléiades 1B, GSD=50 cm). The Toulouse dataset is closely related to our training dataset, as they share similar sensor characteristics and spatial resolutions. On the other hand, the satellite stereo pairs with their specific acquisition geometry, spatial resolution and low signal-to-noise ratio can be considered as *out-of-distribution* datasets. The satellite datasets are therefore appropriate for benchmarking the transferability of our method.

**Metrics.** Evaluation metrics for binary classifiers performance assessment include accuracy, confusion matrix as well as ROC curves, capturing recall at different decision boundaries. Since we target sufficiently separable matching and non-matching feature populations for the subsequent regularization task, we do not privilege a certain threshold. Instead, we estimate the joint probability distribution by sampling matching and non-matching pixel locations at different interval settings (see Fig. 8). A perfect classifier yields matching similarities that are always greater that non-matching ones. We compute joint probability area under the diagonal, denoted as *JP* and the marginal distributions geometric intersection area denoted as *InterA* (see Tab. 2). We also provide *n-pixel* error histograms, 1-, 2- and 3-pixel errors and compare our DeepSim-Nets with MC-CNN acrt

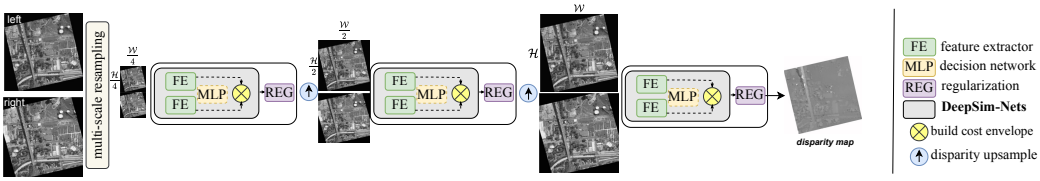



Figure 6. **Multi-Resolution Inference.** We display a *3-scale* iteration. Up to image resolutions $\frac{\mathcal{H}}{8} \times \frac{\mathcal{W}}{8}$, NCC estimates coarse yet robust disparity maps. Moving from resolutions $\frac{\mathcal{H}}{4} \times \frac{\mathcal{W}}{4}$, the down-scaled stereo pairs are iteratively fed to the feature extractor taking one of the three variants: U-Net 32,U-Net Attention, **MS-AFF**. We fill the flexible per-pixel disparity range cost structure either with raw cosine of angles between embeddings (- - →) or with the learnt MLP-based similarities. Then, we upscale the predicted disparity map serving as a predictor for the next iteration.

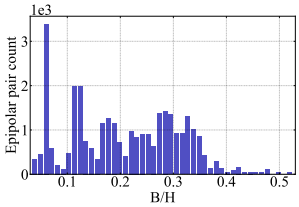



Figure 7. Training dataset base-to-height ratio distribution.

Table 1. Models parameters setting.

| Model | # of params. ($\times 10^3$) |
|---|---|
| MC-CNN [23] | 148 |
| MS-AFF(**Ours**) | 965 |
| U-Net 32 | 7,800 |
| U-Net Attention | 9,500 |
| MLP | 345 |

and PSMNet in Fig. 11 and Tab. 3. Note that PSMNet was trained on a larger aerial dataset acquired on various cities [21], including those used for training our models.

**Ablations.** Three modelling hypothesis are validated: (1) *no_occlusion* loss term contribution, (2) transferability of our model trained on aerial images to a satellite configuration, and (3) the contribution of the MLP-learnt similarity compared to the baseline cosine feature-level similarity.

## 5. Results and discussion

**Similarity learning.** Joint probability maps computed on unseen aerial data are visualised in Fig. 8. The most compact distributions are produced by U-Net 32 and U-Net Attention, which are condensed near 1 for matching similarities (abscissa) and 0 for non-matching ones (ordinate). The MS-AFF distribution qualitatively follows the same trend, but shows a small blob for similarities equal to 0.5, which indicates that some positives and negatives are hard to classify. However, our MS-AFF model yields the highest *JP* for all sampling scenarios (Tab. 2), indicating that the misclassified samples population is negligible. When mixing near and far negative samples (see Fig. 8 (row 3) & Tab. 2 (col. 3)), the variance of our proposed models' joint distributions increases but still outperforms local models by at least 4 %.

Local neighborhood models, including NCC $3 \times 3$, NCC $5 \times 5$, and MCC-CNN acrt, exhibit an increase in *JP* and a decrease in *InterA* as the negative sampling intervals increase, whereas our global ensemblistic models follow the opposite tendency. This occurs because window-based methods enhance the feature's distinctiveness by moving further away from correct matches, where local neighborhood changes drastically and pixel classification becomes easier. On the other hand, our DeepSim-Nets are designed

Table 2. **Quantitative Evaluation.** We evaluate Models' similarity classification performance on unseen aerial Toulouse dataset. Our DeepSim-Nets outperform local methods: MC-CNN acrt, NCC. Our lightweight **MS-AFF** yields the highest *JP* for all sampling scenarios and performs marginally worse than U-Net 32 and U-Net Attention on AUC and *InterA*.

| Sample setting | $\beta_1 = 1, \beta_2 = 4$ | | | $\beta_1 = 2, \beta_2 = 6$ | | | $\beta_1 = 2, \beta_2 = 40$ | | |
|---|---|---|---|---|---|---|---|---|---|
| Aerial Test dataset, $\alpha = 0$ | | | | | | | | | |
| Metrics (%) | JP↑ | InterA↓ | AUC↑ | JP↑ | InterA↓ | AUC↑ | JP↑ | InterA↓ | AUC↑ |
| **Ours:** | | | | | | | | | |
| U-Net 32 + MLP | 85.3 | 23.6 | 92.2 | 88.6 | 15.9 | 95.5 | 87.3 | 19.1 | 89.8 |
| U-Net Att. + MLP | 85.4 | **23.3** | **92.2** | 88.9 | **15.4** | **95.7** | 88.0 | 18.7 | **90.1** |
| MS-AFF + MLP | **86.4** | 23.6 | 91.4 | **89.6** | 15.7 | 95.2 | **88.0** | **18.4** | 89.6 |
| MC-CNN acrt [23] | 78.0 | 33.6 | 85.3 | 82.0 | 25.3 | 89.8 | 83.7 | 24.9 | 87.3 |
| NCC (3×3) | 71.2 | 76.7 | – | 74.8 | 59.1 | – | 77.0 | 45.4 | – |
| NCC (5×5) | 73.2 | 75.4 | – | 76.1 | 60.3 | – | 80.0 | 40.6 | – |

to be distinctive near correct matches, while on large unexplored negative sampling intervals, they may misclassify. The ROC curves in Fig. 9 reveal that our models yield higher recall rates compared to MC-CNN acrt. MS-AFF performs marginally worse than U-Net 32 and U-Net Attention as the AUC demonstrates in Tab. 2. With the model complexity kept in mind, our lightweight MS-AFF shows decent classification results across all examined metrics.

DeepSim-Nets overcome local methods matching ambiguities near correct matches and leverage decent distinctive similarities that are mandatory for the subsequent surface reconstruction. Moreover, although not trained on large negatives sampling offsets, salient matching similarities are obtained when $\beta_1 = 2, \beta_2 = 40$. We also achieve pixel-level separability as well as matching coherence for homogeneous areas.

By explicitly labeling correspondences computed over occlusions as negative samples, we encourage dissimilarity across these regions. This in turn facilitates occlusion detection through simple thresholding of the similarity map. Fig. 10 illustrates this behaviour with MS-AFF trained without and with occlusion self-supervision.

**Surface inference.** We compare DeepSim-Nets' disparities against PSMNet and MC-CNN acrt on *unseen* aerial close-to-distribution (Toulouse) and satellite out-of-distribution stereo pairs (Montpelier, see Tab. 3). We also study the impact of acquisition geometry on the disparity accuracy by looking at the base-to-height ratio ($\frac{B}{H}$). On Toulouse dataset, we show that our models outperform

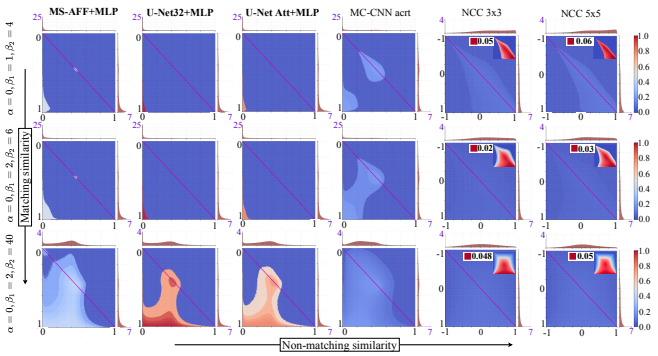



Figure 8. **Classifiers Accuracy**. From top to bottom, we enlarge the negatives' sampling interval defined by $\beta_1$ and $\beta_2$. We estimate joint as well as marginal matching/non-matching similarity distributions for the **three** variants of DeepSim-Nets, MC-CNN acrt and NCC with $3 \times 3$ and $5 \times 5$ window sizes. For visualization purposes, we normalize all joint distributions *w.r.t* the maximum distribution value in each row and display equalized thumbnails for NCC distributions. These maps give us insights on the matching and non-matching "pixels" separability of our binary classifiers. Our DeepSim-Nets (first 3 columns) accumulate almost all observations under the diagonal. MC-CNN acrt and NCC misclassify and render high variance maps.

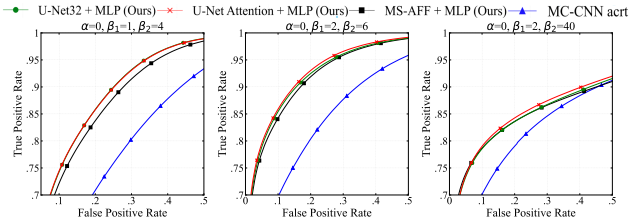



Figure 9. **ROC Curve Analysis**. We represent ROC curves for *three* negatives sampling intervals (i.e scenarios) defined by offsets $\beta_1$ and $\beta_2$ *w.r.t* ground truth locations ($\alpha = 0$). Our ensemble models yield the lowest False Positive Rates (FPR) for different recall rates compared to MC-CNN acrt for all sampling scenarios.

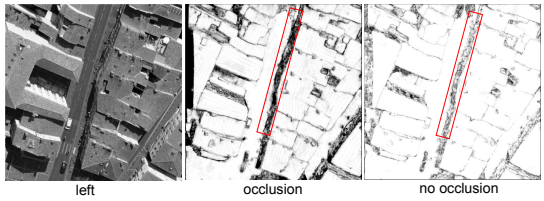



Figure 10. **Contribution of The Occlusion Term**. When the occlusion-specific loss term is activated, similarity values drop drastically in occluded regions □.

PSMNet and MC-CNN acrt in occlusion-free regions almost for all examined metrics (Fig. 11 & Tab. 3). Nonetheless, PSMNet recovers precisely buildings' outlines while our method may render poor edge shapes, especially near occlusions (see Fig. 1). This said, PSMNet has the tendency to smooth surfaces and occasionally add high frequency low amplitude artefacts (see Fig. 1), while we faithfully reproduce rooftop details (see Fig. 12). For $\frac{B}{H} = 0.2$, U-Net Attention slightly outperforms U-Net 32 and MS-AFF on both

Table 3. **Ablation Study.** Accuracy assessment on aerial/satellite unseen datasets. Statistics are calculated on difference maps between models' induced disparities and ground truth in non-occluded areas. For the aerial Toulouse dataset, we examine varying $\frac{B}{H}$. For Montpellier, Pléiades 1B dataset, no $\frac{B}{H}$ selection is done. $\mu$ is the mean absolute difference, $\sigma$ is the standard deviation, *NMAD* is the normalized median absolute deviation. $D_1$, $D_2$ and $D_3$ are 1-, 2- and 3-pixel error rates (%) respectively.

| | Method | $\frac{B}{H}$ | $\mu\downarrow$ | $\sigma\downarrow$ | *NMAD*$\downarrow$ | $D_1\downarrow$ | $D_2\downarrow$ | $D_3\downarrow$ |
|---|---|---|---|---|---|---|---|---|
| | Toulouse aerial 8cm GSD dataset | | | | | | | |
| OURS | MS-AFF cos | 0.2 | 0.42 | 0.98 | 0.12 | 7.10 | 3.43 | 2.40 |
| OURS | U-Net 32 cos | 0.2 | 0.40 | **0.96** | 0.11 | 6.61 | 3.26 | 2.30 |
| OURS | U-Net -Attention cos | 0.2 | **0.40** | 0.96 | **0.11** | **6.40** | **3.19** | **2.28** |
| OURS | MS-AFF+MLP | 0.2 | 0.39 | 0.95 | **0.11** | 6.30 | 3.22 | 2.28 |
| OURS | U-Net 32+MLP | 0.2 | 0.39 | **0.95** | 0.11 | 6.18 | 3.13 | 2.24 |
| OURS | U-Net -Attention+MLP | 0.2 | **0.39** | 0.95 | 0.11 | **5.97** | **3.10** | **2.24** |
| | MC-CNN acrt [23] | | 0.52 | 1.14 | 0.14 | 9.88 | 5.23 | 3.60 |
| | PSMNet [3] | | 0.49 | 1.00 | 0.15 | 8.66 | 4.60 | 3.01 |
| OURS | MS-AFF cos | 0.48 | 1.18 | 1.61 | **0.37** | **31.37** | 13.62 | 9.09 |
| OURS | U-Net 32 cos | 0.48 | 1.19 | 1.58 | 0.38 | 32.66 | 13.38 | 8.67 |
| OURS | U-Net -Attention cos | 0.48 | **1.18** | **1.58** | 0.38 | 32.34 | **13.23** | **8.60** |
| OURS | MS-AFF+MLP | 0.48 | 1.27 | **1.49** | **0.40** | **39.24** | **14.60** | 8.65 |
| OURS | U-Net 32+MLP | 0.48 | 1.28 | 1.50 | 0.42 | 40.35 | 14.75 | 8.44 |
| OURS | U-Net -Attention+MLP | 0.48 | **1.27** | 1.50 | 0.42 | 39.56 | 14.77 | **8.41** |
| | MC-CNN acrt [23] | | 2.10 | 1.96 | 0.99 | 60.31 | 39.73 | 23.45 |
| | PSMNet [3] | | 1.34 | 1.65 | 0.44 | 39.10 | 16.68 | 10.42 |
| OURS | MS-AFF cos | All | 0.70 | 1.30 | 0.21 | 15.97 | 7.15 | 4.85 |
| OURS | U-Net 32 cos | All | 0.69 | 1.28 | 0.21 | 16,15 | 6.97 | 4.63 |
| OURS | U-Net -Attention cos | All | **0.68** | **1.28** | **0.20** | **15,88** | **6.86** | **4.59** |
| OURS | MS-AFF+MLP | All | 0.71 | **1.25** | 0.22 | 18.20 | 7.33 | 4.57 |
| OURS | U-Net 32+MLP | All | 0.72 | 1.26 | 0.22 | 18.59 | 7.35 | 4.50 |
| OURS | U-Net -Attention+MLP | All | **0.71** | 1.25 | **0.22** | **18.14** | **7.32** | **4.47** |
| | MC-CNN acrt [23] | | 1.09 | 1.66 | 0.29 | 29.12 | 17.65 | 10.74 |
| | PSMNet [3] | | 0.79 | 1.33 | 0.24 | 19.18 | 8.78 | 5.57 |
| | Montpellier Pléaides 1B 50 cm GSD satellite dataset | | | | | | | |
| OURS | MS-AFF cos | – | 0.75 | 0.98 | **0.27** | 20.07 | 7.9 | 4.20 |
| OURS | U-Net 32 cos | – | 0.75 | 1.00 | 0.27 | **19.94** | 7.86 | 4.18 |
| OURS | U-Net -Attention cos | – | **0.75** | **0.97** | 0.28 | 20.27 | **7.70** | **3.98** |
| OURS | MS-AFF+MLP | – | 1.27 | 1.26 | 0.50 | 45.31 | 17.56 | 8.72 |
| OURS | U-Net 32+MLP | – | 1.17 | 1.22 | 0.46 | 39.84 | 15.82 | 7.81 |
| OURS | U-Net -Attention+MLP | – | 1.20 | 1.26 | 0.49 | 40.87 | 17.13 | 8.56 |
| | MC-CNN acrt [23] | | 0.84 | 1.13 | 0.30 | 23.13 | 10.14 | 5.54 |
| | PSMNet [3] | | 1.02 | 1.18 | 0.40 | 32.27 | 13.78 | 7.00 |

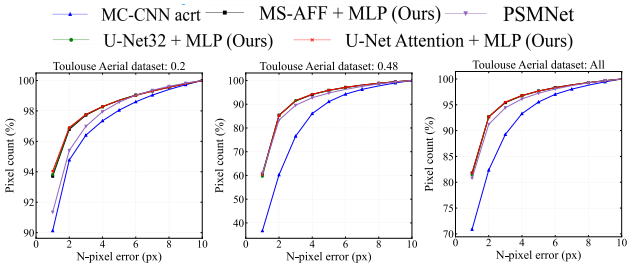
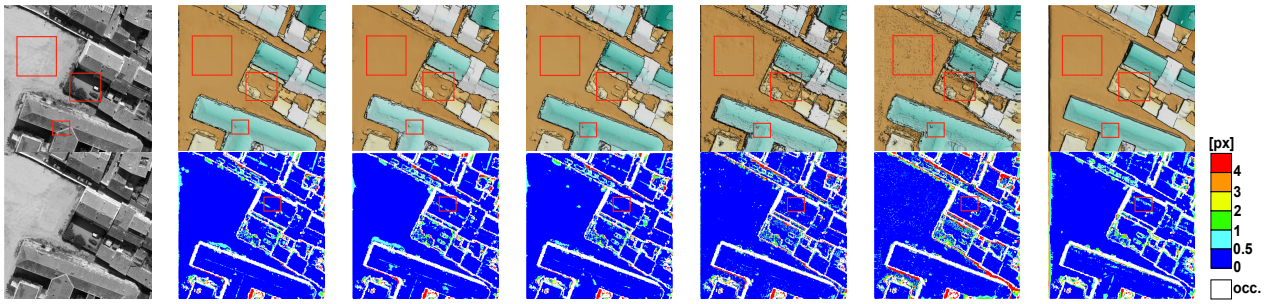



Figure 11. **Quantitative Results**. Error histograms evaluated on occlusion-free areas. Our models outperform PSMNet and MC-CNN acrt for different $\frac{B}{H}$ settings.

feature-based *cosine* and MLP-based similarities. The MLP decision module gain is about 0.5 % for all models. This shows that feature modules provide sufficiently generic representations, deployable for downstream tasks such as 3D reconstruction. As larger $\frac{B}{H}$ (i.e., 0.48) are not represented in the training data (see Fig. 7), the performance of our models coupled with the MLP similarity deteriorates. PSMNet




(a) Epipolar Pair. (b) **Ours**:MS-AFF. (c) **Ours**:U-Net 32. (d) **Ours**:U-Net Att. (e) MC-CNN acrt. (f) NCC5 $\times$ 5. (g) PSMNet.

Figure 12. **Disparity Predictions on Aerial Images**. (top: colored and grey-shaded disparity maps, bottom: difference maps *w.r.t* ground truth). Here, we evaluate the entire setting (feature extractor + MLP). Conversely to MC-CNN acrt and NCC, planar surfaces with poor contextual information (big □) are recovered best by our models. Shadows are handled well by DeepSimNets and PSMNet (middle □). PSMNet renders consistent reconstructions on buildings boundaries and near occlusions but ignores tiled roof patterns that are recovered by almost all similarity-driven models (small □).

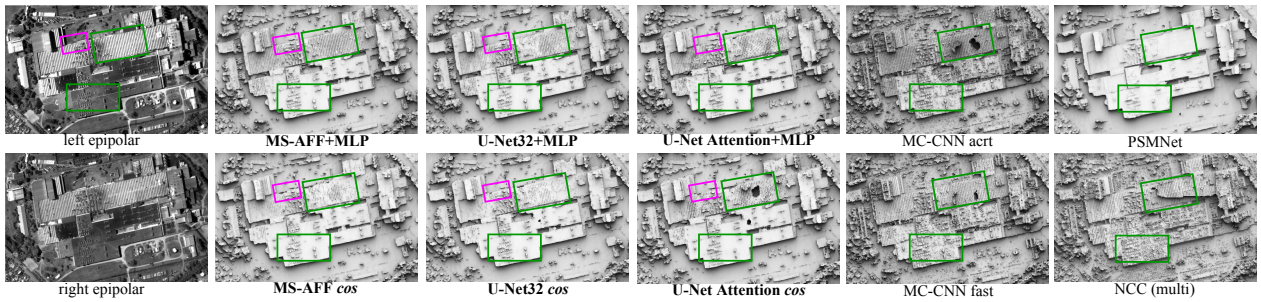



Figure 13. **Disparity Predictions on Satellite Images**. Grey-shaded disparity maps capturing the performance of tested methods on unseen WV3 stereo pairs over Buenos Aires. While local neighborhood classifiers: MC-CNN(fast&acrt),NCC(multiple windows) fail to reconstruct fine-grain building rooftops' details □, our models recover such high frequency details. PSMNet acts as a low-pass filter. The MLP-learnt similarities recover buildings' outlines □ missed by raw-cosine (*cos*) similarities.

is slightly better compared to our best performing architecture MS-AFF+MLP. As the MLP clearly specializes to seen $\frac{B}{H}$, feature representations alone remain powerful and expressive: MS-AFF cosine outperforms PMSNet by 7.73 % on $D_1$. For $\frac{B}{H} = 0.48$, MS-AFF with or without MLP is more robust to acquisition geometric changes than the rest of the models. When no particular $\frac{B}{H}$ configuration is privileged, cosine-based similarities are more advantageous in presence of varying $\frac{B}{H}$.

On the unseen WV-3 stereo pairs, the DeepSim-Nets reconstruct buildings' details and boundaries more faithfully and with less regularization, whereas PSMNet outputs fuzzy buildings and erases fine details (Fig. 13). Local methods (NCC, MC-CNN) produce noisy surfaces with disparity jumps on repetitive rooftop patterns. On the unseen Pléiades 1B stereo pairs, we evaluated our models and compared them with the MC-CNN acrt trained using our sampling scheme. Although our best-performing model was U-Net 32+MLP, it was outperformed by the MC-CNN acrt (see Tab. 3). Interestingly, deactivating the MLP improved our model's transferability and the accuracy of the disparity maps. In contrast to PSMNet, which merged buildings and their shadows into a single entity, our method accurately classified shadows as ground features. Notably, our U-Net 32 cosine model showed a significant 3.19% improvement in the $D_1$ metric when compared to MC-CNN acrt.

## 6. Conclusion

In this study, we have presented several variants of DeepSim-Nets for learning stereo-correspondence, which outperform standard hybrid methods on all examined metrics. Our networks can allocate sets of pixels in epipolar geometry and learn similarities simultaneously, overcoming the locality constraint and leveraging more global, context-aware, and transferable similarity cues. We have designed a sample mining scheme that improves deep feature extractors and enables occlusion detection in our models through contrastive training. Our flexible and lightweight MS-AFF model is designed to fit large multi-scale iterative dense matching pipelines, and generalizes well to unseen aerial and satellite stereo pairs.

## 7. Acknowledgements

This work was funded by Thales. We thank the AI4Geo project for the HPC resources and the Pléiades 1B dataset.

## References


[1] Konstantinos Batsos, Changjiang Cai, and Philippos Mordohai. CBMV: A coalesced bidirectional matching volume for disparity estimation. In *CVPR*, 2018. 2

[2] Marc Bosch, Kevin Foster, Gordon A. Christie, Sean Wang, Gregory D. Hager, and Myron Z. Brown. Semantic stereo for incidental satellite images. In *WACV*, 2018. 1

[3] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *CVPR*, 2018. 1, 2, 7

[4] Zhuoyuan Chen, Xun Sun, Liang Wang, Yinan Yu, and Chang Huang. A deep visual correspondence embedding model for stereo matching costs. In *ICCV*, 2015. 2, 3, 4

[5] Yimian Dai, Fabian Gieseke, Stefan Oehmcke, Yiquan Wu, and Kobus Barnard. Attentional feature fusion. In *WACV*, 2021. 2, 3

[6] Shivam Duggal, Shenlong Wang, Wei-Chiu Ma, Rui Hu, and Raquel Urtasun. Deeppruner: Learning efficient stereo matching via differentiable patchmatch. In *ICCV*, 2019. 2

[7] Philipp Fischer, Alexey Dosovitskiy, Eddy Ilg, Philip Häusser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *ICCV*, 2015. 2

[8] Xufeng Han, Thomas Leung, Yangqing Jia, Rahul Sukthankar, and Alexander C. Berg. Match-net: Unifying feature and metric learning for patch-based matching. In *CVPR*, 2015. 2, 3, 4

[9] Wilfried Hartmann, Silvano Galliani, Michal Havlena, Konrad Schindler, and Luc Van Gool. Learned multi-patch similarity. In *ICCV*, 2017. 2

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[11] Heiko Hirschmüller. Stereo processing by semiglobal matching and mutual information. In *IEEE TPAMI*, 2008. 1, 2

[12] Alex Kendall, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry. End-to-end learning of geometry and context for deep stereo regression. In *ICCV*, 2017. 2

[13] Ozan Oktay, Jo Schlemper, Loïc Le Folgoc, Matthew C. H. Lee, Mattias P. Heinrich, Kazunari Misawa, Kensaku Mori, Steven G. McDonagh, Nils Y. Hammerla, Bernhard Kainz, Ben Glocker, and Daniel Rueckert. Attention u-net: Learning where to look for the pancreas. 2018. 1, 2, 3

[14] Sonali Patil, Bharath Comandur, Tanmay Prakash, and Avinash C. Kak. A new stereo benchmarking dataset for satellite images. In *arXiv*, 2019. 1

[15] Marc Pierrot-Deseilligny and Nicolas Paparoditis. Multiresolution and optimization-based image matching approach : an application to surface reconstruction from SPOT6-HRS stereo imagery. In *ISPRS Archives*, 2006. 1, 2, 4, 5

[16] Zhibo Rao, Mingyi He, Zhidong Zhu, Yuchao Dai, and Renjie He. Bidirectional guided attention network for 3-d semantic detection of remote sensing images. In *IEEE GRSS*, 2021. 2

[17] Damien Robert, Bruno Vallet, and Loic Landrieu. Learning multi-view aggregation in the wild for large-scale 3d semantic segmentation. In *CVPR*, 2022. 3

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI*, 2015. 1, 3

[19] Sebastien Roy and Ingemar J. Cox. A maximum-flow formulation of the n-camera stereo correspondence problem. In *ICCV*, 1998. 2

[20] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015. 4

[21] Tend Wu, Bruno Vallet, Marc Pierrot-Deseilligny, and Ewelina Rupnik. A new stereo dense matching benchmark dataset for deep learning. In *ISPRS Annals*, 2021. 1, 5, 6

[22] Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. In *CVPR*, 2015. 2

[23] Jure Zbontar and Yann LeCun. Stereo matching by training a convolutional neural network to compare image patches. In *JMLR*, 2016. 2, 3, 4, 6, 7

[24] Feihu Zhang, Victor Adrian Prisacariu, Ruigang Yang, and Philip H. S. Torr. Ga-net: Guided aggregation net for end-to-end stereo matching. In *CVPR*, 2019. 2

[25] Iman Zolanvari, Susana Ruano, Aakanksha Rana, Alan Cummins, Aljosa Smolic, Rogerio Da Silva, and Morteza Rahbar. Dublincity: Annotated lidar point cloud and its applications. 2019. 5