Proposal Flow

Bumsub Ham, Minsu Cho, Cordelia Schmid, Jean Ponce

Proposal Flow

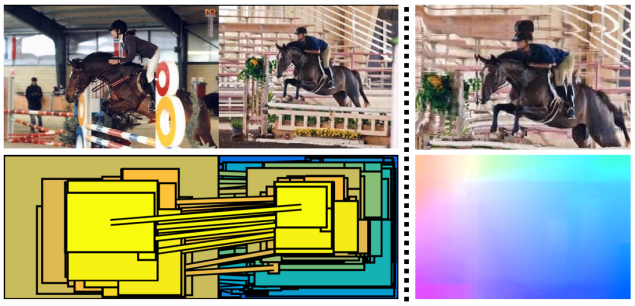
Bumsub Ham[1,*,†] Minsu Cho[1,*,†] Cordelia Schmid[1,‡] Jean Ponce[2,†]
[1]Inria [2]École Normale Supérieure / PSL Research University

Abstract

Finding image correspondences remains a challenging problem in the presence of intra-class variations and large changes in scene layout, typical in scene flow computation. We introduce a novel approach to this problem, dubbed proposal flow, that establishes reliable correspondences using object proposals. Unlike prevailing scene flow approaches that operate on pixels or regularly sampled local regions, proposal flow benefits from the characteristics of modern object proposals, that exhibit high repeatability at multiple scales, and can take advantage of both local and geometric consistency constraints among proposals. We also show that proposal flow can effectively be transformed into a conventional dense flow field. We introduce a new dataset that can be used to evaluate both general scene flow techniques and region-based approaches such as proposal flow. We use this benchmark to compare different matching algorithms, object proposals, and region features within proposal flow with the state of the art in scene flow. This comparison, along with experiments on standard datasets, demonstrates that proposal flow significantly outperforms existing scene flow methods in various settings.

1. Introduction

Classical approaches to finding correspondences across images are designed to handle scenes that contain the same objects with moderate view point variations in applications such as stereo matching [42, 46], optical flow [23, 45, 51], and wide-baseline matching [41, 53]. *Scene flow* methods, such as SIFT Flow [35] for example, on the other hand, are designed to handle a much higher degree of variability in appearance and scene layout. They have proven useful for many tasks such as scene recognition, image registration, semantic segmentation, and image editing and synthesis [20, 29, 35, 53, 56]. In this context, however, appearance variations may confuse similarity measures for local region matching, and prohibit the use of strong geometric constraints (e.g., epipolar geometry, limited disparity range). Existing approaches to scene flow are thus easily distracted by scene elements specific to individual objects and image-specific details (e.g., background, texture, occlusion, clutter). The motivation of our work is that robust region correspondences and scene flow require focusing on regions containing prominent objects and scene elements rather than clutter and distracting details.



(a) Region-based scene flow. (b) Flow field.

Figure 1. Proposal flow generates a reliable scene flow between similar images by establishing geometrically consistent correspondences between object proposals. (a) Region-based scene flow by matching object proposals. (b) Color-coded dense flow field generated from the region matches, and image warping using the flow. (Best viewed in color.)

To address these issues, we introduce an approach to scene flow computation, called *proposal flow*, that establishes region correspondences using object proposals and their geometric relations (Fig. 1). Unlike previous scene flow algorithms [4, 20, 22, 25, 29, 35, 44, 48, 49, 53, 56], that use regular grid structures for local region generation and matching, we leverage a large number of multi-scale object proposals [1, 24, 40, 50, 57], as now widely used in object detection [19, 27]. The proposed approach establishes region correspondences by exploiting their visual features and geometric relations in an efficient manner, and generates a region-based scene flow composed of object proposal matches. We also show that the proposal flow can be effectively transformed into a conventional dense flow field. Finally, we introduce a new dataset that can

*indicates equal contribution

†WILLOW project-team, Département d'Informatique de l'Ecole Normale Supérieure, ENS/Inria/CNRS UMR 8548.

‡LEAR project-team, Inria Grenoble Rhône-Alpes, Laboratoire Jean Kuntzmann, CNRS, Univ. Grenoble Alpes, France.

be used to evaluate both general scene flow techniques and region-based approaches such as proposal flow. We use this benchmark to compare different matching algorithms, object proposals, and region features within proposal flow with the state of the art in scene flow. This comparison, along with experiments on standard datasets, demonstrates that proposal flow significantly outperforms existing scene flow methods in various settings.

2. Related work

Correspondence problems involve a broad range of topics beyond the scope of this paper. Here we briefly describe the context of our approach, and only review representative work pertinent for ours. Classical approaches to stereo matching and optical flow estimate pixel-level dense correspondences between two nearby images of the same scene [23, 41, 42]. While advances in invariant feature detection and description have revolutionized object recognition and reconstruction in the past 15 years, research on image matching and alignment between images have long been dominated by instance matching with the same scene and objects [18]. Unlike these, several recent approaches focus on handling images containing different scenes and objects. Graph-based matching algorithms [10, 14] attempt to find category-level feature matches by leveraging a flexible graph representation of images, but they commonly handle sparsely sampled or detected features due to their computational complexity. Inspired by classic optical flow algorithms, Liu *et al*. pioneered the idea of dense correspondences across different scenes, and proposed the SIFT Flow [35] algorithm that uses a multi-resolution image pyramid together with a hierarchical optimization technique for efficiency. Kim *et al*. [29] extended the approach by inducing a multi-scale regularization with a hierarchically connected pyramid of grid graphs. More recently, Long *et al*. [36] have investigated the effect of pretrained ConvNet features on scene flows, and Bristow *et al*. [4] have proposed an exemplar-LDA approach that improves the performance of scene flows. Despite differences in graph construction, optimization, and similarity computation, existing scene flow approaches share grid-based regular sampling and spatial regularization: The appearance similarity is defined at each region or pixel on (a pyramid of) regular grids, and spatial regularization is imposed between neighboring regions in the pyramid models [29, 35]. In contrast, our work builds on generic object proposals with diverse spatial supports [1, 24, 40, 50, 57], and uses an irregular form of spatial regularization based on co-occurrence and overlap of the proposals. We show that the use of local regularization with object proposals yields substantial gains in generic region matching and scene flow, in particular, when handling images with significant clutter and intra-class variations.

Object proposals [1, 24, 40, 50, 57] have originally been developed for object detection, where they are used to reduce the search space as well as false alarms. They are now an important component in many state-of-the-art detection pipelines [19, 27]. Despite their success on object detection and segmentation, they have seldom been used in matching tasks [9, 26]. In particular, while Cho *et al*. [9] have shown that object proposals are useful for object discovery due to their high repeatability on salient part regions, the use of object proposals has never been throughly investigated in scene flow computation. The proposal flow approach proposed in this paper is a first step in this direction, and we explore how the choice of object proposals, matching algorithms, and features affects matching robustness and accuracy.

Contributions. The contributions of this paper are threefold: (i) We introduce the proposal flow approach to establishing robust region correspondences between related, but not identical scenes using object proposals. (ii) We introduce a benchmark for scene flow that can be used to evaluate both general scene flow algorithms and region matching methods such as proposal flow. (iii) We demonstrate the advantage of proposal flow over state-of-the-art scene flow methods through extensive experimental evaluations.

3. Proposal flow

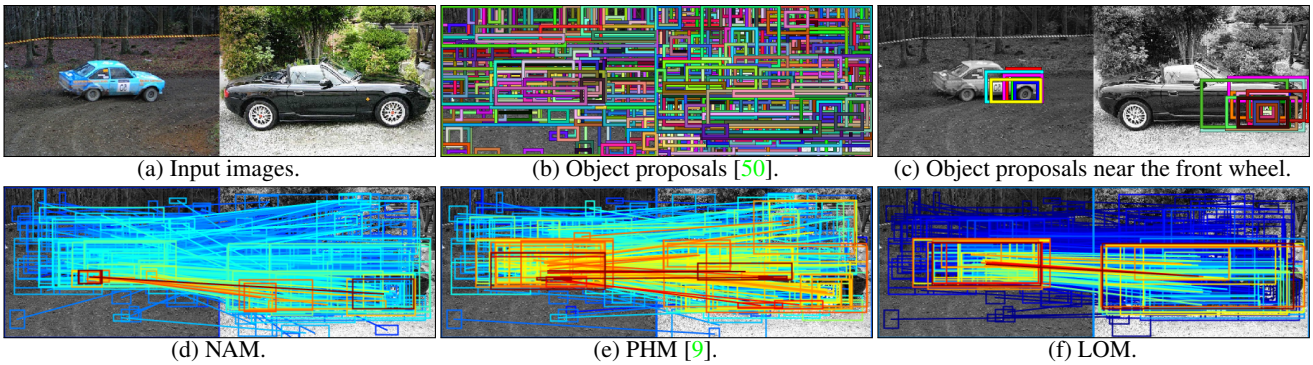
Proposal flow can use any type of object proposals [1, 24, 40, 50, 57] as candidate regions for matching two images of related scenes. In this section, we introduce a probabilistic model for region matching, and describe three matching strategies including two baselines and a new one using local regularization. We then describe our approach to generating a dense flow field from region matches.

3.1. A Bayesian model for region confidence

Let us suppose that two sets of object proposals $\mathcal{R}$ and $\mathcal{R}'$ have been extracted from images $\mathcal{I}$ and $\mathcal{I}'$ (Fig. 2(a-b)). A proposal r in $\mathcal{R}$ is an image region $r = (f, s)$ with appearance feature f and spatial support s. The appearance feature represents a visual descriptor for the region (e.g., SPM [31] , HOG [11], ConvNet [30]), and the spatial support describes the set of all pixel positions in the region, that forms a rectangular box in this work. Given the data $\mathcal{D} = (\mathcal{R}, \mathcal{R}')$, we wish to estimate a posterior probability of the event $r \mapsto r'$ meaning that proposal r in $\mathcal{R}$ matches proposal r' in $\mathcal{R}'$:

$$p(r \mapsto r' \mid \mathcal{D}) = p(f \mapsto f')p(s \mapsto s' \mid \mathcal{D}), \qquad (1)$$

where we decouple the probabilities of appearance and spatial support matching, and assume that appearance matching is independent of $\mathcal{D}$. In practice, the appearance term $p(f \mapsto f')$ is simply computed from a similarity between



(a) Input images. (b) Object proposals [50]. (c) Object proposals near the front wheel.

(d) NAM. (e) PHM [9]. (f) LOM.

Figure 2. **Top:** (a-b) Two images and their object proposals [50]. (c) Multi-scale object proposals contain the same object or parts, but they are not consistently generated across different images. **Bottom:** In contrast to NAM (d), PHM [9] (e) and LOM (f) both exploit geometric consistency, which regularizes proposal flow. In particular, LOM imposes local smoothness on offsets between neighboring regions, avoiding the problem of using a global consensus on the offset in PHM [9]. The matching score is color-coded for each match (red: high, blue: low). The HOG descriptor [11] is used for appearance matching in this example. (Best viewed in color.)

feature descriptors f and f', and the geometric consistency term $p(s \mapsto s' \mid \mathcal{D})$ is evaluated by comparing the spatial supports s and s' in the context of the given data $\mathcal{D}$, as described in the next section. We set the posterior probability as a matching score and assign the best match $\phi(r)$ for each proposal in $\mathcal{R}$:

$$\phi(r) = \underset{r' \in \mathcal{R}'}{\operatorname{argmax}}\, p(r \mapsto r' \mid \mathcal{D}). \tag{2}$$

Using a slight abuse of notation, if $(f', s') = \phi(f, s)$, we will write $f' = \phi(f)$ and $s' = \phi(s)$.

3.2. Geometric matching strategies

We now introduce three matching strategies, using different geometric consistency terms $p(s \mapsto s' \mid \mathcal{D})$.

Naive appearance matching (NAM). A straightforward way of matching regions is to use a uniform distribution for the geometric term so that

$$p(r \mapsto r' \mid \mathcal{D}) \propto p(f \mapsto f'). \tag{3}$$

NAM considers appearance only, and does not reflect any geometric relationship among regions (Fig. 2(d)).

Probabilistic Hough matching (PHM). The matching algorithm in [9] can be expressed in our model as follows. First, a three-dimensional location vector (center position and scale) is extracted from the spatial support s. We denote it by a function γ. An offset space $\mathcal{X}$ is defined as a feasible set of offset vectors between $\gamma(s)$ and $\gamma(s')$: $\mathcal{X} = \{\gamma(s) - \gamma(s') \mid r \in \mathcal{R}, r' \in \mathcal{R}'\}$. The geometric consistency term $p(s \mapsto s' \mid \mathcal{D})$ is then defined as

$$p(s \mapsto s' \mid \mathcal{D}) = \sum_{x \in \mathcal{X}} p(s \mapsto s' \mid x) p(x \mid \mathcal{D}), \tag{4}$$

which assumes that $p(s \mapsto s' \mid x, \mathcal{D}) = p(s \mapsto s' \mid x)$. Here, $p(s \mapsto s' \mid x)$ measures an offset consistency between $\gamma(s) - \gamma(s')$ and x by a Gaussian kernel in the three-dimensional offset space. From this model, PHM substitutes $p(x \mid \mathcal{D})$ with a generalized Hough transform score:

$$h(x \mid \mathcal{D}) = \sum_{(r, r') \in \mathcal{D}} p(f \mapsto f') p(s \mapsto s' \mid x). \tag{5}$$

which aggregates individual votes for offset x, from *all* possible matches in $\mathcal{D} = \mathcal{R} \times \mathcal{R}'$. Hough voting imposes a spatial regularizer on matching by taking into account a global consensus on the corresponding offset [33, 39]. However, it often suffers from background clutter that distracts the global voting process (Fig. 2(e)).

Local offset matching (LOM). Here we propose a new method to overcome this drawback of PHM [9] and obtain more reliable correspondences. Object proposals often contain a large number of distracting outlier regions from background clutter, and are not perfectly repeatable even for corresponding object or parts across different images (Fig. 2(c)). The global Hough voting in PHM has difficulties with such outlier regions (background clutter). In contrast, LOM optimizes a translation and scale offset for each proposal by exploiting only neighboring proposals. This local approach substantially alleviates the effect of outlier regions in matching as will be demonstrated by our experimental results.

The main issue is how to estimate a reliable offset for each proposal r in a robust manner without any information about objects and their locations. One way would be to find the corresponding region of the region r through a multi-scale sliding window search in $\mathcal{I}'$ as in object detection [16], but this is expensive. Instead, we assume that nearby regions have similar offsets. For each region r, we

first define its neighborhood $\mathcal{N}(r)$ as the regions with overlapping spatial support:

$$\mathcal{N}(r) = \{\hat{r} \mid s \cap \hat{s} \neq \emptyset, \hat{r} \in \mathcal{R}\}. \quad (6)$$

Using an initial correspondence $\phi(r)$, determined by the best match according to the appearance term, each neighboring region $\hat{r}$ is assigned its own offset, and all of them form a set of neighbor offsets:

$$\mathcal{X}(r) = \{\gamma(\hat{s}) - \gamma(\phi(\hat{s})) \mid \hat{r} \in \mathcal{N}(r)\}. \quad (7)$$

From this set of neighbor offsets, we estimate a local offset x_r^* for the region r by the geometric median [37][1]:

$$x_r^* = \underset{x \in \mathbb{R}^3}{\operatorname{argmin}} \sum_{y \in \mathcal{X}(r)} \|x - y\|_2 , \quad (8)$$

which can be globally optimized by Weiszfeld's algorithm [6] using a form of iteratively re-weighted least squares. Based on the local offset x_r^* optimized for each region, we define the geometric consistency function:

$$g(s \mapsto s'|\mathcal{D}) = p(s \mapsto s'|x_r^*) \sum_{\hat{r} \in \mathcal{N}(r)} p(\hat{f} \mapsto \phi(\hat{f})), \quad (9)$$

which means that r in $\mathcal{R}$ is likely to match with r' in $\mathcal{R}'$ if their offset is close to the local offset x_r^*, and r has many neighboring matches with a high appearance fidelity.

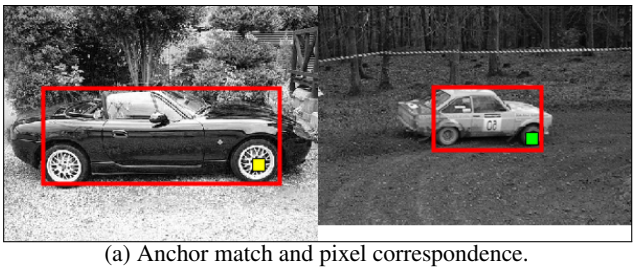
By using $g(s \mapsto s'|\mathcal{D})$ as a proxy for $p(s \mapsto s'|\mathcal{D})$, LOM imposes local smoothness on offsets between neighboring regions. In particular, it regularizes offsets within a local neighborhood that incorporates an overlap relationship between spatial supports of regions. This local regularization avoids a common problem with PHM, where the matching results often depend on a few strong matches. The use of local offsets optimized for each proposal effectively suppresses matches between clutter regions, while favoring matches between regions that contain objects rather than object parts (Fig. 2(f)).

3.3. Flow field generation

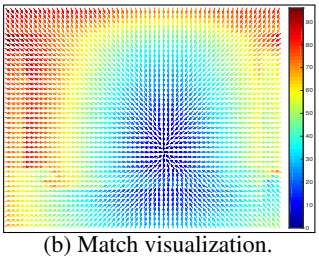
The proposal flow gives a set of region correspondences between images, but it can be easily transformed into a conventional flow field. Let p denote a pixel in image $\mathcal{I}$ (yellow point in Fig. 3(a)). For each pixel p, its neighborhood is defined as the region in which it lies, *i.e.*, $\mathcal{N}(p) = \{r \in \mathcal{R} : p \in r\}$. We define an anchor match $(r^*, \phi(r^*))$ as the region correspondence that has the highest matching score among neighboring regions (red boxes in Fig. 3(a)) where

$$r^* = \underset{r \in \mathcal{N}(p)}{\operatorname{argmax}}\, p(r \mapsto \phi(r) \mid \mathcal{D}). \quad (10)$$

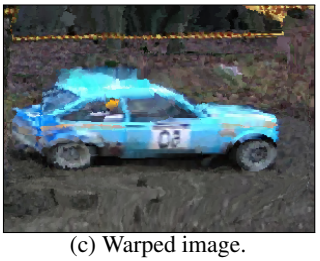
[1]We found that the centroid and mode of the offset vectors in three-dimensional offset space show worse performance than the geometric median. This is because the neighboring regions may include clutters. Clutter causes incorrect neighbor offsets, but the geometric median is robust to outliers [17], providing a reliable local offset.



(a) Anchor match and pixel correspondence.



(b) Match visualization.



(c) Warped image.

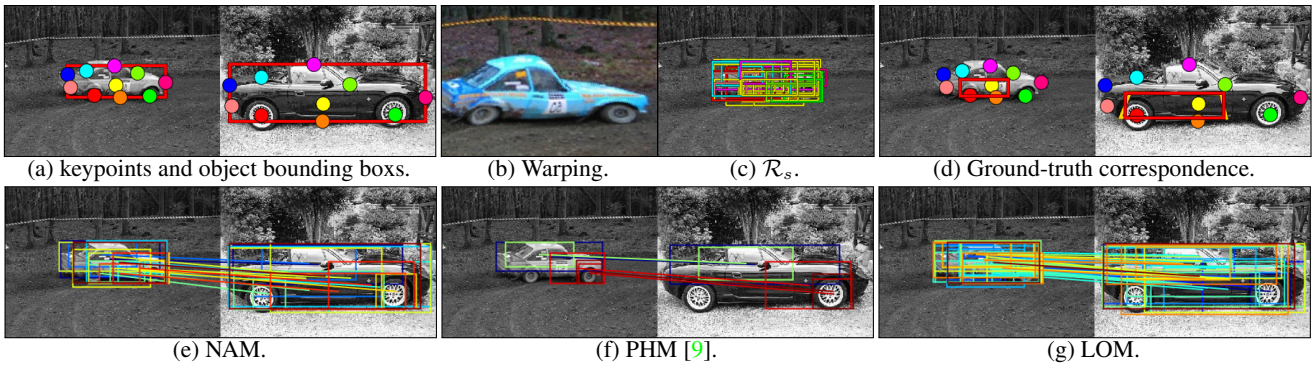
Figure 3. Flow field generation. (a) For each pixel (yellow point), its anchor match (red boxes) is determined. The correspondence (green point) is computed by the transformed coordinate with respect to the position and size of the anchor match. (b) Based on the flow field, (c) the right image is warped to the left image. The warped object shows visually similar shape to the one in the left image. The LOM method is used for region matching with the object proposals [40] and the HOG descriptor [11]. (Best viewed in color.)

Note that the anchor match contains information on translation and scale changes between objects. Using the geometric relationships between the pixel p and its anchor match $(r^*, \phi(r^*))$, a correspondence p' in $\mathcal{I}'$ (green point in Fig. 3(a)) is obtained by linear interpolation.

The matching score for each correspondence is set to the value of its anchor match. When p and q in $\mathcal{I}$ are matched to the same pixel p' in $\mathcal{I}'$, we select the match with the highest matching score and delete the other one. Finally, joint image filtering [21] is applied under the guidance of the image $\mathcal{I}$ to interpolate the flow field in places without correspondences. Figure 3(b-c) shows examples of the estimated flow field and corresponding warping result between two images: Using the dense correspondences, we warp all pixels in the right image to the left image. Our approach using the anchor match aligns semantic object parts well while handling translation and scale changes between objects.

4. RSF: A new dataset for scene flow evaluation

Current research on scene flow lacks an appropriate benchmark with dense ground truth correspondences. Conventional optical flow benchmarks (e.g., Middlebury [2] and MPI-SINTEL [5]) do not feature within-class variations, and ground truth for generic scene flow is difficult to capture due to its intrinsically semantic nature, manual annotation being extremely labor intensive and somewhat subjective. All existing approaches are thus evaluated only with sparse



(a) keypoints and object bounding boxs. (b) Warping. (c) $\mathcal{R}_s$. (d) Ground-truth correspondence.

(e) NAM. (f) PHM [9]. (g) LOM.

Figure 4. (a-d) Generating ground-truth regions and evaluating correct matches. (a) Using keypoint annotations, dense correspondences between images are established using warping [3, 13]. (b) Based on the dense correspondences, all pixels in the left image are warped to the right image. (c) We assume that true matches exist only between the regions near the object bounding box, and thus an evaluation is done with the regions in this subset of object proposals. (d) For each object proposal (red box in the left image), its ground truth is generated automatically by the dense correspondences: We use a tight rectangle (red box in the right image) of the region formed by the warped object proposal (yellow box in the right image) as a ground-truth correspondence. (e-g) Examples of correct matches: The numbers of correct matches are 16, 5, and 38 for NAM (e), PHM [9] (f), and LOM (g), respectively. Matches with IoU score greater than 0.5 are considered as correct in this example. (Best viewed in color.)

ground truth or in an indirect manner (e.g. mask transfer accuracy) [4, 29, 35, 44, 48, 56]. Such benchmarks only evaluate a small number of matches, that occur at ground-truth keypoints or around mask boundaries in a point-wise manner. To address this issue, we propose in this section a new benchmark for scene flow, dubbed *RSF* for *region-based scene flow*, built using ground-truth object bounding boxes and keypoint annotations, (Fig. 4(a-b)), and propose new evaluation metrics for region-based scene flow. Note that while designed for region-based methods, our benchmark can be used to evaluate any scene flow technique. As will be seen in our experiments, it provides a reasonable (if approximate) ground truth for dense correspondences across similar scenes without an extremely expensive annotation campaign. As shown in the following sections, comparative evaluations on this dataset are also good predictors for performance on other tasks and datasets, further justifying its use of RSF. In the following, we describe our ground-truth generation process, evaluation criteria, and datasets. The benchmark data and code will be made available online.

4.1. Ground-truth correspondence generation

We assume that true matches only exist within object bounding boxes. Let us assume two sets of keypoint annotations at positions k_i and k'_i in $\mathcal{I}$ and $\mathcal{I}'$, respectively, with $i = 1, \dots, m$. Assuming the objects present in the images and their parts may undergo shape deformation, we use thin plate splines (TPS) [3, 13] to interpolate the sparse keypoints. Namely, the ground truth is approximated from sparse correspondences using TPS warping.

For each region, its ground truth match is generated as follows. We assume that true matches only exist between a subset of regions, *i.e.*, regions around object bounding boxes (Fig. 4(c)): $\mathcal{R}_s = \{r \mid |b \cap r| \,/\, |r| \geq 0.75, r \in \mathcal{R}\}$ where b denotes an object bounding box in $\mathcal{I}$, and $|r|$ indicates the area of a region r. For each region $r \in \mathcal{R}_s$, the four vertices of the rectangle are warped to the corresponding ones in $\mathcal{I}'$ by the TPS mapping function. The region formed by the warped points is a correspondence of region r. We fit a tight rectangle for this region and set it as a ground-truth correspondence for the region r (Fig. 4(d)).

4.2. Evaluation criteria

We introduce two evaluation metrics for region matching performance in terms of *region matching precision* and *match retrieval accuracy*. Basically, the metrics build on the intersection over union (IoU) score between r's correspondence $\phi(r)$ and its ground truth $r^\star$:

$$\text{IoU}(\phi(r), r^\star) = |\phi(r) \cap r^\star| \,/\, |\phi(r) \cup r^\star|. \qquad (11)$$

For region matching precision, we propose the probability of correct region (PCR) metric[2] where region r is correctly matched to its ground truth $r^\star$ if $1 - \text{IoU}(\phi(r), r^\star) > \tau$ (e.g., Fig. 5(a) top). We measure the PCR metric while varying the IoU threshold τ from 0 to 1. For match retrieval accuracy, we propose the average IoU of k-best matches (mIoU@k) according to the matching score (e.g., Fig. 5(a) bottom). We measure the mIoU@k metric while increasing the number of top matches k. These two metrics exhibit two important characteristics in matching: the PCR reveals the accuracy of overall assignment, and the mIoU@k shows

[2]This region-based metric is developed from a conventional point-based metric, the probability of correct keypoint (PCK) [54]. In the case of pixel-based flow, the PCK can be adopted instead.

the reliability of match confidence scores that is crucial in match selection.

4.3. Dataset construction

To generate our dataset, we start from the benchmark for sparse matching of Cho *et al.* [8], which consists of 5 object classes (Face, Car, Motorbike, Duck, WineBottle) with 10 keypoint annotations for each image. Note that these images contain more clutter and intra-class variation than existing datasets for scene flow evaluation, *e.g.*, images with tightly cropped objects or similar background [29, 44, 56]. We exclude the Face class where the number of generated object proposals is not enough to evaluate matching accuracy. The other classes are split into sub-classes[3] according to the viewpoint or background clutters. We obtain a total of 10 sub-classes. Given these images and regions, we generate ground-truth data between all possible image pairs within each class.

5. Experiments

5.1. Experimental details

Object proposals. We evaluate four state-of-the-art object proposal methods: EdgeBox (EB) [57], multi-scale combinatorial grouping (MCG) [1], selective search (SS) [50], and randomized prim (RP) [40]. In addition, we consider three baseline proposals [24]: Uniform sampling (US), Gaussian sampling (GS), and sliding window (SW). See [24] for more details. For fair comparison, we use 1,000 proposals for all the methods. To control the number of proposals, we use the proposal score provided for EB, MCG, and SS. For RP, we randomly select among the proposals.

Feature descriptors and similarity. We evaluate three popular feature descriptors: SPM [31], HOG [11], and ConvNet [30]. For SPM, dense SIFT features [38] are extracted every 4 pixels and each descriptor is quantized into a 1,000 word codebook [47]. For each region, a spatial pyramid pooling [31] is used with 1×1 and 3×3 pooling regions. We compute the similarity between SPM descriptors by the χ^2 kernel. HOG features are extracted with 8×8 cells and 31 orientations, then whitened. For ConvNet features, we use each output of the 5 convolutional layers in AlexNet [30] that is pre-trained on the ImageNet dataset [12]. For HOG and ConvNet, the dot product is used as a similarity metric.

5.2. Proposal flow components

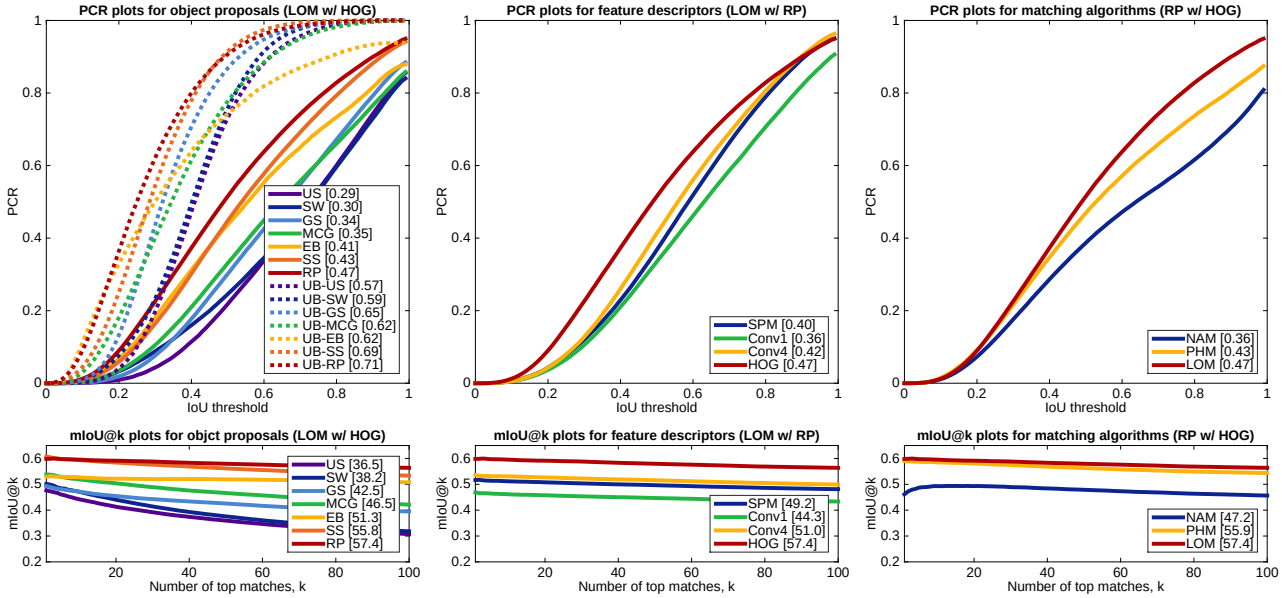
We use the RSF benchmark in this section to compare three versions of proposal flow (NAM, PHM, LOM), combined with different matching algorithms and various object proposals [1, 24, 40, 50, 57], and features [11, 30, 31]. Figure 4(e-g) shows a qualitative comparison between region matching algorithms on a pair of images and depicts correct matches found by each variant of proposal flow. In this example, at the IoU threshold 0.5, the numbers of correct matches are 16, 5, and 38 for NAM, PHM [9], and LOM, respectively. This shows that PHM may give worse performance than even NAM when much clutter exists in background. In contrast, the local regularization in LOM alleviates the effect of such clutter.

Figure 5 summarizes the matching and retrieval performance on average for all object classes with a variety of combination of object proposals, feature descriptors, and matching algorithms. Figure 5(a) compares different types of object proposals with fixed matching algorithm and feature descriptor (LOM w/ HOG). RP shows the best matching precision and retrieval accuracy among the object proposals. An upper bound on precision is measured for each object proposal method, that is the best performance we can achieve with the proposal method. The upper bound (UB) plots show that RP generates more consistent regions than other proposal methods, and is adequate for region matching. RP shows higher matching precision than other proposals especially when the IoU threshold is low. The evaluation results for different features (LOM w/ RP) are shown in Fig. 5(b). The HOG descriptor gives the best performance in matching and retrieval. Among ConvNet features, the fourth and first convolutional layers (Conv4 and Conv1) show the best and worst performance, respectively, while other layers perform similar to SPM. This confirms the finding in [55], which shows that Conv4 gives the best matching performance among ImageNet-trained ConvNet features. Figure 5(c) compares the performance of different matching algorithms (RP w/ HOG), and shows that LOM outperforms others in matching as well as retrieval. Figure 5(d and e) shows the area under curve (AuC) for PCR and mIoU@k plots, respectively. This suggests that combining of LOM, RP, and HOG performs best in both metrics. In Table 1, we show per class AuCs for PCR plots (LOM w/ RP and HOG). From this table, we can see that 1) higher matching precision is achieved with objects having a similar pose (e.g., mot(S) vs. mot(M)), 2) performance decreases for deformable object matching (e.g., duck(S) vs. car(S)), and 3) matching precision can increase drastically by eliminating background clutters (e.g., win(w/o C) vs. win(w/ C)).

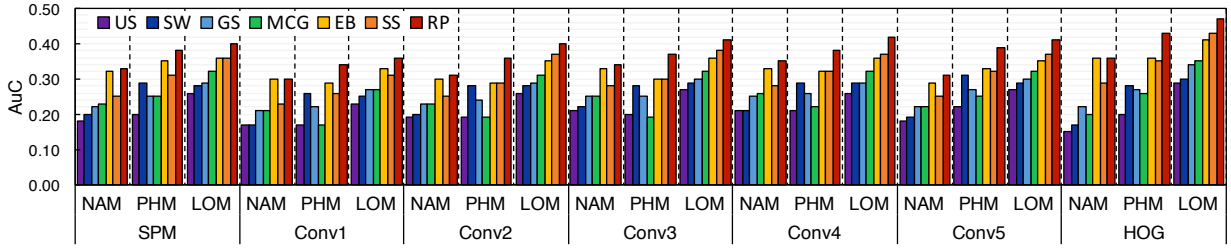
5.3. Flow field

To compare our method with state-of-the-art scene flow methods, we compute a conventional flow field from our proposal flows (Sec. 3.3), and evaluate image alignment between all pairs of images in each subset. We test four object proposal methods (MCG, EB, SS, RP) with HOG de-

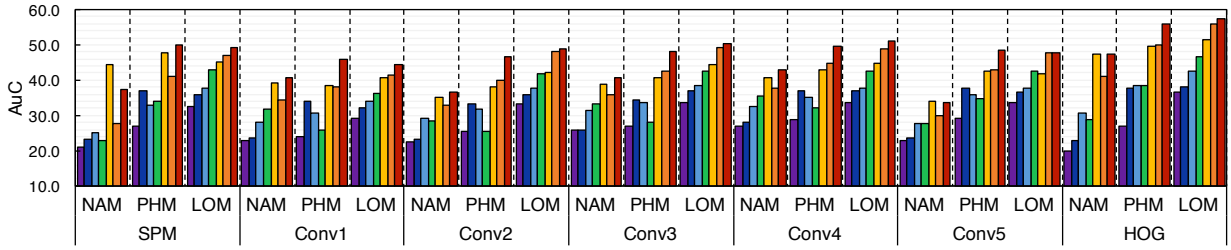
[3]They are car (S), (G), (M), duck (S), motorbike (S), (G), (M), wine bottle (w/o C), (w/ C), (M), where (S) and (G) denote side and general viewpoints, respectively. (C) stands for background clutter, and (M) denotes mixed viewpoints (side + general) for car and motorbike classes and a combination of images in wine bottle (w/o C + w/ C) for the wine bottle class. The dataset has 10 images for each class, thus 100 images in total.



(a) Comparison of object proposals. (b) Comparison of feature descriptors. (c) Comparison of matching algorithms.



(d) AuCs for PCR curves.



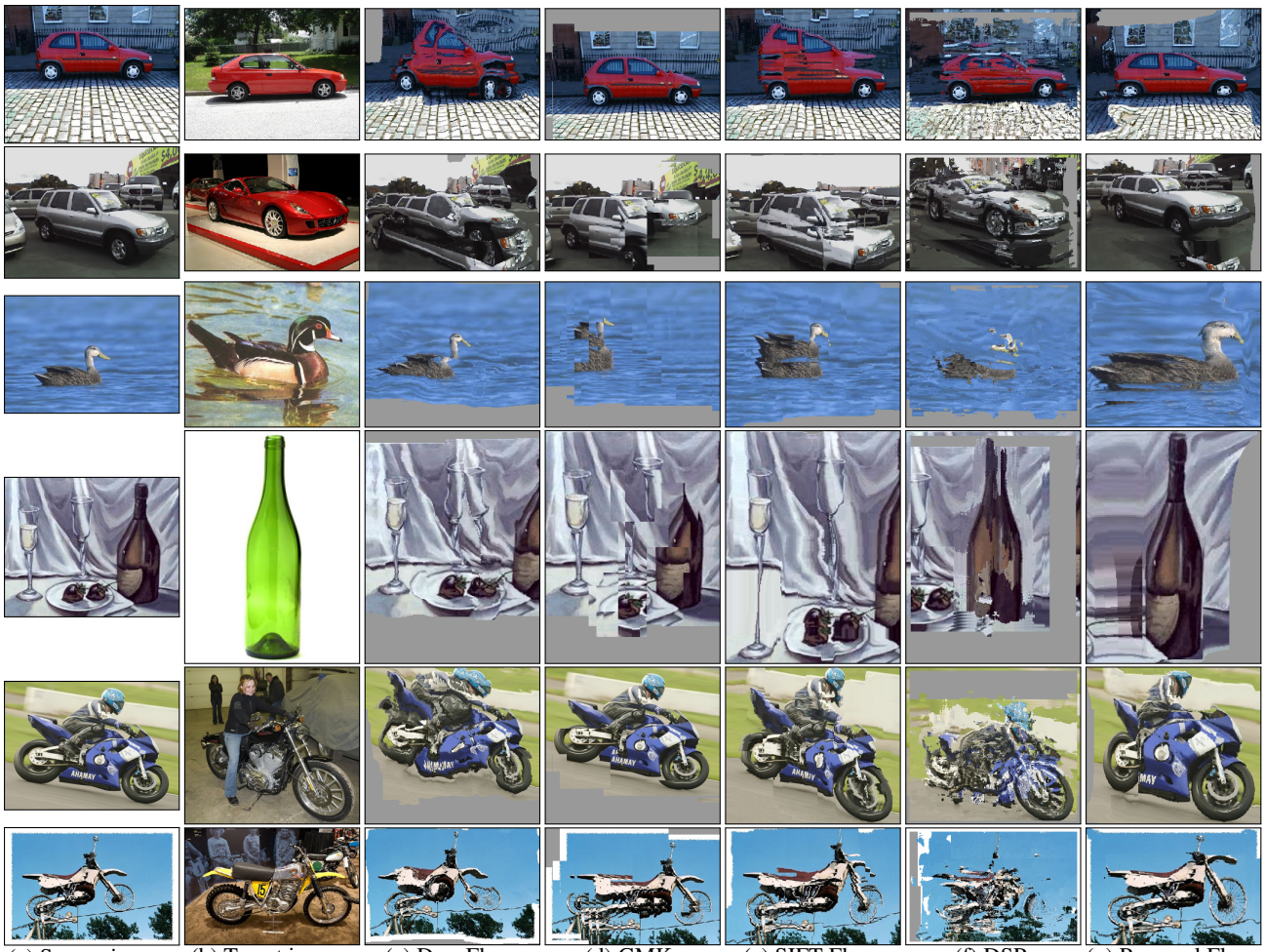
(e) AuCs for mIoU@k curves.

Figure 5. RSF benchmark evaluation on (a-c) region matching precision (top, PCR plots) and match retrieval accuracy (bottom, mIoU@k plots), and (d-e) AuCs for different combinations of object proposals, features, and matching algorithms: (a) Evaluation for LOM with HOG [11], (b) evaluation for LOM with RP [40], (c) evaluation for RP with HOG [11], (d) AuCs for PCR curves, and (e) AuCs for mIoU@k curves. The AuC is shown in the legend. (Best viewed in color.)

Table 1. AuC performance for PCR plots (LOM w/ RP and HOG).

Methods	car(S)	car(G)	car(M)	duck(S)	mot(S)	mot(G)	mot(M)	win(w/o C)	win(w/ C)	win(M)	Avg.
LOM	0.61	0.50	0.45	0.50	0.42	0.40	0.35	0.69	0.30	0.47	0.47
Upper bound	0.75	0.69	0.69	0.72	0.70	0.70	0.67	0.80	0.68	0.73	0.71

scriptors. For an evaluation metric, we use PCK ($\alpha = 0.1$) between warped keypoints and ground-truth ones [36, 54]. Ground-truth keypoints are deemed to be correctly predicted if they lie within $\alpha \cdot \max(h, w)$ pixels of the predicted points for α in $[0, 1]$, where h and w are the height and width of the object bounding box, respectively. Table 2 shows the average PCK over all object classes. In our benchmark, all versions of proposal flow significantly outperform SIFT Flow [35], DSP [29], and DeepFlow [45]. LOM with SS or RP outperforms other combination of matching and pro-



(a) Source image. (b) Target image. (c) DeepFlow. (d) GMK. (e) SIFT Flow. (f) DSP. (g) Proposal Flow.

Figure 6. Examples of dense flow field. (a-b) Source images are warped to the target images using the dense correspondences estimated by (c) DeepFlow [45], (d) GMK [14], (e) SIFT Flow [35], (f) DSP [29], and (g) Proposal Flow w/ LOM, RP and HOG.

Table 2. PCK ($\alpha = 0.1$) comparison for dense flow field.

Methods	MCG [1]	EB [57]	SS [50]	RP [40]
NAM	0.46	**0.50**	0.52	0.53
PHM	0.48	0.45	0.55	0.54
LOM	**0.49**	0.44	**0.56**	**0.55**
DeepFlow [45]	0.20			
GMK [14]	0.27			
SIFT Flow [35]	0.38			
DSP [29]	0.37			

Table 3. Matching accuracy on the Caltech-101 dataset.

Proposals	Methods	LT-ACC	IoU	LOC-ERR
SS [50]	NAM	0.68	0.44	0.41
	PHM	0.74	0.48	0.32
	LOM	**0.78**	**0.50**	**0.25**
RP [40]	NAM	0.70	0.44	0.39
	PHM	0.75	0.48	0.31
	LOM	**0.78**	**0.50**	0.26
DeepFlow [45]		0.74	0.40	0.34
GMK [14]		0.77	0.42	0.34
SIFT Flow [35]		0.75	0.48	0.32
DSP [29]		0.77	0.47	0.35

posal methods, which coincides with the results in Sec 5.2. Figure 6 gives a qualitative comparison with the state of the art on these examples. The better alignment found by proposal flow here is typical of our experiments. Specifically, proposal flow is robust to translation and scale changes between objects.

Matching results on Caltech-101. We also evaluate our approach on the Caltech-101 dataset [15]. Following the experimental protocol in [29], we randomly select 15 pairs of images for each object class, and evaluate matching accuracy with three metrics: label transfer accuracy (LT-ACC) [34], the IoU metric, and the localization error (LOC-ERR) of corresponding pixel positions. For LT-ACC, we transfer the class label of one image to the other using dense correspondences, and count the number of correctly labeled pixels. Similarly, the IoU score is measured between the transferred label and ground truth. Table 3 compares quantitatively the matching accuracy of proposal flow to the state

(a) Source image. (b) Target image. (c) DSP. (d) Proposal Flow. (e) Source mask. (f) Target mask. (g) DSP. (h) Proposal Flow.

Figure 7. Examples of dense flow field on PASCAL parts. (a-b) Source images are warped to the target images using the dense correspondences estimated by (c) DSP [29] and (d) Proposal Flow w/ LOM, SS and HOG. (e-f) Similarly, annotated part segments for the source images are warped to the target images using the dense correspondences computed by (g) DSP and (h) Proposal Flow w/ LOM, SS and HOG. (Best viewed in color.)

of the art. It shows that proposal flow using LOM outperforms other approaches, especially for the IoU score and the LOC-ERR of dense correspondences. Note that compared to LT-ACC, these metrics evaluate the matching quality for the foreground object, separate from irrelevant scene clutter. Our results verify that proposal flow focuses on regions containing objects rather than scene clutter and distracting details, enabling robust image matching against outliers.

Matching results on PASCAL parts. We use the dataset provided by [56] where the images are sampled from the PASCAL part dataset [7]. We first measure part matching accuracy using human-annotated part segments. For this experiment, we measure the weighted IoU score between transferred segments and ground truths, with weights determined by the pixel area of each part (Table 4). To evaluate alignment accuracy, we measure the PCK metric

Table 4. Matching accuracy on the PASCAL VOC classes.

Methods	IoU	PCK
Congealing [32]	0.38	0.11
RASL [43]	0.39	0.16
CollectionFlow [28]	0.38	0.12
DSP [29]	0.39	**0.17**
NAM	0.35	0.13
PHM	0.39	**0.17**
LOM	**0.41**	**0.17**

($\alpha = 0.05$) using keypoint annotations for the 12 rigid PASCAL classes [52] (Table. 4). We use the same set of images as in the part matching experiment. Proposal flow has an advantage over existing approaches on images that contain cluttering elements (e.g., background, instance-specific texture, occlusion), but in this dataset [56], such elements are confined to only a small portion of the images, compared to our benchmark and the Caltech-101 datasets [15]. This may be a reason that, for PCK performance, our approach with SS [50] gives similar results to other methods. Figure 7 visualize the part matching results.

For more details about our experiments, see our supplementary material.

6. Discussion

We have presented a robust region-based scene flow method, called proposal flow, and showed that it can effectively be mapped onto pixel-wise dense correspondences. We have also introduced the RSF dataset for scene flow, and shown that it provides a reasonable benchmark for scene flow evaluation without extremely expensive manual annotation of full ground-truth. RSF can be used to evaluate region-based scene flow methods and even pixel-based ones, and experiments with RSF demonstrate that proposal flow substantially outperforms existing scene flow methods. Experiments with Caltech and the VOC parts datasets further validate these results.

Acknowledgments. This work was supported by the ERC grants VideoWorld and Allegro, and the Institut Universitaire de France.

References

[1] P. Arbelaez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *CVPR*, 2014. 1, 2, 6, 8

[2] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *IJCV*, 92(1):1–31, 2011. 4

[3] F. L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE TPAMI*, (6):567–585, 1989. 5

[4] H. Bristow, J. Valmadre, and S. Lucey. Dense semantic correspondence where every pixel is a classifier. In *ICCV*, 2015. 1, 2, 5

[5] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*. 2012. 4

[6] R. Chandrasekaran and A. Tamir. Open questions concerning weiszfeld's algorithm for the fermat-weber location problem. *Mathematical Programming*, 44(1-3):293–295, 1989. 4

[7] X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, et al. Detect what you can: Detecting and representing objects using holistic models and body parts. In *CVPR*, 2014. 9

[8] M. Cho, K. Alahari, and J. Ponce. Learning graphs to match. In *ICCV*, 2013. 6

[9] M. Cho, S. Kwak, C. Schmid, and J. Ponce. Unsupervised object discovery and localization in the wild: Part-based matching using bottom-up region proposals. In *CVPR*, 2015. 2, 3, 5, 6

[10] M. Cho and K. M. Lee. Progressive graph matching: Making a move of graphs via probabilistic voting. In *CVPR*. 2

[11] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 2, 3, 4, 6, 7

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*. 6

[13] G. Donato and S. Belongie. Approximate thin plate spline mappings. In *ECCV*, 2002. 5

[14] O. Duchenne, A. Joulin, and J. Ponce. A graph-matching kernel for object categorization. In *ICCV*, 2011. 2, 8

[15] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *TPAMI*, 28(4):594–611, 2006. 8, 10

[16] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008. 3

[17] P. T. Fletcher, S. Venkatasubramanian, and S. Joshi. Robust statistics on riemannian manifolds via the geometric median. In *CVPR*, 2008. 4

[18] D. A. Forsyth and J. Ponce. Computer vision: A modern approach (2nd edition). *Computer Vision: A Modern Approach*, 2011. 2

[19] R. Girshick. Fast R-CNN. In *ICCV*, 2015. 1, 2

[20] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski. Non-rigid dense correspondence with applications for image enhancement. *ACM TOG*, 30(4):70, 2011. 1

[21] B. Ham, M. Cho, and J. Ponce. Robust image filtering using joint static and dynamic guidance. In *CVPR*, 2015. 4

[22] T. Hassner, V. Mayzels, and L. Zelnik-Manor. On SIFTs and their scales. In *CVPR*, 2012. 1

[23] B. K. Horn and B. G. Schunck. Determining optical flow: A retrospective. *Artificial Intelligence*, 59(1):81–87, 1993. 1, 2

[24] J. Hosang, R. Benenson, P. Dollár, and B. Schiele. What makes for effective detection proposals? *TPAMI*, 2015. 1, 2, 6

[25] J. Hur, H. Lim, C. Park, and S. C. Ahn. Generalized deformable spatial pyramid: Geometry-preserving dense correspondence estimation. 2015. 1

[26] H. Jiang. Matching bags of regions in RGBD images. In *CVPR*, 2015. 2

[27] H. Kaiming, Z. Xiangyu, R. Shaoqing, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *ECCV*, 2014. 1, 2

[28] I. Kemelmacher-Shlizerman and S. M. Seitz. Collection flow. In *CVPR*, 2012. 9

[29] J. Kim, C. Liu, F. Sha, and K. Grauman. Deformable spatial pyramid matching for fast dense correspondences. In *CVPR*, 2013. 1, 2, 5, 6, 7, 8, 9

[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 2, 6

[31] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006. 2, 6

[32] E. G. Learned-Miller. Data driven image models through continuous joint alignment. *TPAMI*, 28(2):236–250, 2006. 9

[33] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *IJCV*, 77(1-3):259–289, 2008. 3

[34] C. Liu, J. Yuen, and A. Torralba. Nonparametric scene parsing via label transfer. *TPAMI*, 33(12):2368–2382, 2011. 8

[35] C. Liu, J. Yuen, and A. Torralba. SIFT flow: Dense correspondence across scenes and its applications. *TPAMI*, 33(5):978–994, 2011. 1, 2, 5, 7, 8

[36] J. L. Long, N. Zhang, and T. Darrell. Do convnets learn correspondence? In *NIPS*, 2014. 2, 7

[37] H. P. Lopuhaa and P. J. Rousseeuw. Breakdown points of affine equivariant estimators of multivariate location and covariance matrices. *The Annals of Statistics*, pages 229–248, 1991. 4

[38] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 6

[39] S. Maji and J. Malik. Object detection using a max-margin hough transform. In *CVPR*, 2009. 3

[40] S. Manen, M. Guillaumin, and L. Van Gool. Prime object proposals with randomized Prim's algorithm. In *ICCV*, 2013. 1, 2, 4, 6, 7, 8

[41] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and vision computing*, 22(10):761–767, 2004. 1, 2

[42] M. Okutomi and T. Kanade. A multiple-baseline stereo. *TPAMI*, 15(4):353–363, 1993. 1, 2

[43] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma. Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images. *TPAMI*, 34(11):2233–2246, 2012. 9

[44] W. Qiu, X. Wang, X. Bai, Z. Tu, et al. Scale-space SIFT flow. In *WACV*, 2014. 1, 5, 6

[45] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. Deepmatching: Hierarchical deformable dense matching. *ArXiv e-prints*, 2015. 1, 7, 8

[46] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. 2011. 1

[47] K. Tang, A. Joulin, L.-J. Li, and L. Fei-Fei. Co-localization in real-world images. In *CVPR*, 2014. 6

[48] M. Tau and T. Hassner. Dense correspondences across scenes and scales. *TPAMI (To appear)*, 2015. 1, 5

[49] E. Trulls, I. Kokkinos, A. Sanfeliu, and F. Moreno-Noguer. Dense segmentation-aware descriptors. In *CVPR*, 2013. 1

[50] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013. 1, 2, 3, 6, 8, 10

[51] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. Deepflow: Large displacement optical flow with deep matching. In *ICCV*, 2013. 1

[52] Y. Xiang, R. Mottaghi, and S. Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *WACV*, 2014. 10

[53] H. Yang, W.-Y. Lin, and J. Lu. Daisy filter flow: A generalized discrete approach to dense correspondences. In *CVPR*, 2014. 1

[54] Y. Yang and D. Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE TPAMI*, 35(12):2878–2890, 2013. 5, 7

[55] S. Zagoruyko and N. Komodakis. Learning to compare image patches via convolutional neural networks. In *CVPR*, 2015. 6

[56] T. Zhou, Y. Jae Lee, S. X. Yu, and A. A. Efros. FlowWeb: Joint image set alignment by weaving consistent, pixel-wise correspondences. In *CVPR*, 2015. 1, 5, 6, 9, 10

[57] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014. 1, 2, 6, 8