Generic 3D Representation via Pose Estimation and Matching

Amir R. Zamir[1](✉), Tilman Wekel[1], Pulkit Agrawal[2], Colin Wei[1], Jitendra Malik[2], and Silvio Savarese[1]

[1] Stanford University, Stanford, USA
zamir@cs.stanford.edu
[2] University of California, Berkeley, USA
http://3Drepresentation.stanford.edu/

Abstract. Though a large body of computer vision research has investigated developing generic semantic representations, efforts towards developing a similar representation for 3D has been limited. In this paper, we learn a generic 3D representation through solving a set of foundational proxy 3D tasks: object-centric camera pose estimation and wide baseline feature matching. Our method is based upon the premise that by providing supervision over a set of carefully selected foundational tasks, generalization to novel tasks and abstraction capabilities can be achieved. We empirically show that the internal representation of a multi-task ConvNet trained to solve the above core problems generalizes to novel 3D tasks (e.g., scene layout estimation, object pose estimation, surface normal estimation) without the need for fine-tuning and shows traits of abstraction abilities (e.g., cross modality pose estimation).

In the context of the core supervised tasks, we demonstrate our representation achieves state-of-the-art wide baseline feature matching results without requiring apriori rectification (unlike SIFT and the majority of learnt features). We also show 6DOF camera pose estimation given a pair local image patches. The accuracy of both supervised tasks come comparable to humans. Finally, we contribute a large-scale dataset composed of object-centric street view scenes along with point correspondences and camera pose information, and conclude with a discussion on the learned representation and open research questions.

Keywords: Generic vision · Representation · Descriptor learning · Pose estimation · Wide-baseline matching · Street view

1 Introduction

Supposed an image is given and we are interested in extracting some 3D information from it, such as, the scene layout or the pose of the visible objects. One

Electronic supplementary material The online version of this chapter (doi:10.1007/978-3-319-46487-9_33) contains supplementary material, which is available to authorized users.

B. Leibe et al. (Eds.): ECCV 2016, Part III, LNCS 9907, pp. 535–553, 2016.
DOI: 10.1007/978-3-319-46487-9_33

potential approach would be to annotate a dataset for every single desired problem and train a fully supervised system for each (i.e., supervised learning). This is undesirable as an annotated dataset for each problem would be needed as well as the fact that the problems would be treated independently. In addition, unlike semantic annotations such as, object labels, certain annotations in 3D are cumbersome to collect and often require special sensors (imagine manually annotating exact pose of an object or surface normals). An alternative approach is to develop a system with a rather generic perception that can conveniently generalize to novel tasks. In this paper, we take a step towards developing a generic 3D perception system that (1) can solve novel 3D problems without fine-tuning, and (2) is capable of certain abstract generalizations in the 3D context (e.g., reason about pose similarity between two drastically different objects).

But, how could one learn such a generalizable system? Cognitive studies suggest living organisms can perform cognitive tasks for which they have not received supervision by supervised learning of other foundational tasks [28,45,51]. Learning the relationship between visual appearance and changing the vantage point (self-motion) is among the first visual skills developed by infants and play a fundamental role in developing other skills, e.g., depth perception. A classic experiment [28] showed a kitten that was deprived from self-motion experienced fundamental issues in 3D perception, such as failing to understand depth when placed on the Visual Cliff [22]. Later works [45] argued this finding was not, at least fully, due to motion intentionality and the supervision signal of self-motion was indeed a crucial elements in learning basic visual skills. What these studies essentially suggest are: (1) by receiving supervision on a certain proxy task (in this case, self-motion), other tasks (depth understanding) can be solved sufficiently without requiring an explicit supervision, (2) some vision tasks are more foundational than others (e.g., self-motion perception vs depth understanding).

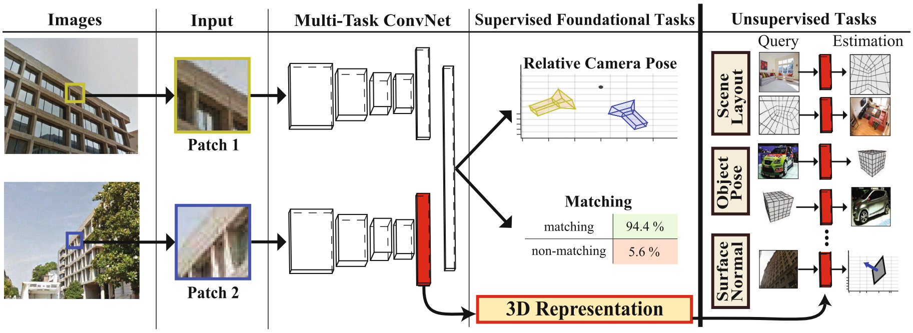

Fig. 1. Learning a generic 3D representation: we develop a supervised joint framework for camera pose estimation and wide baseline matching. We then show the internal representation of this framework can be used as a 3D representation generalizable to various 3D prediction tasks.

Inspired by the above discussion, we develop a supervised framework where a ConvNet is trained to perform 6DOF camera pose estimation. This basic task allows learning the relationship between an arbitrary change in the viewpoint and the appearance of an object/scene-point. One property of our approach is performing the camera pose estimation in a object/scene-centric manner: the training data is formed of image bundles that show the *same point of an object/scene* while the camera moves around (i.e., it fixates - see the Fig. 2(c)). This is different from existing video+metadata datasets [20], the problem of Visual Odometry [20,42], and recent works on ego-motion estimation [4,29], where in the training data, the camera moves independent of the scene. Our object/scene-centric approach is equivalent to allowing a learner to focus on a physical point while moving around and observing how the appearance of that particular point transforms according to viewpoint change. Therefore, the learner receives an additional piece of information that the observed pixels are indeed showing the same object, giving more information about how the element looks under different viewpoints and providing better grounds for learning visual encoding of an observation. Infants also explore object-motion relationships [51] in a similar way as they hold an object in hand and observe it from different views.

Our dataset also provides supervision for the task of wide baseline matching, defined as identifying if two images/patches are showing the same point regardless of the magnitude of viewpoint change. Wide baseline matching is also an important 3D problem and is closely related to object/scene-centric camera pose estimation: to identify whether two images could be showing the same point despite drastic changes in the appearance, an agent could learnt how viewpoint change impacts the appearance. Therefore, we perform our supervised training in a multi-task manner to simultaneously solve for both wide baseline matching and pose estimation. This has the advantage of learning a single representation that encodes both problems. In experiments Sect. 4.1, we show it is possible to have a single representation solving both problems without a performance drop compared to having two dedicate representations. This provides practical computational and storage advantages. Also, training ConvNets using multiple tasks/losses is desirable as it has been shown to be better regularized [23,58,63].[1]

We train the ConvNet (siamese structure with weight sharing) on patch pairs extracted from the training data and use the last FC vector of one siamese tower as the generic 3D representation (see Fig. 1). We will empirically investigate if this representation can be used for solving novel 3D problems (we evaluated on scene layout estimation, object pose estimation, surface normal estimation), and whether it can perform any 3D abstraction (we experimented on cross category pose estimation and relating the pose of synthetic geometric elements to images).

Dataset: We developed an object-centric dataset of street view scenes from the cities of Washington DC, NYC, San Francisco, Paris, Amsterdam, Las Vegas,

[1] Though visual matching/tracking is also one of early developed cognitive skills [10], we are unware of any studies investigating its foundational role in developing visual perception. Therefore, we presume (and empirically observe) that the generality of our 3D representation is mostly attributed to the camera pose estimation component.

and Chicago, augmented with *camera pose* information and *point correspondences* (with >half a billion training data points). We release the dataset, trained models, and an online demo at http://3Drepresentation.stanford.edu/.

Novelty in the Supervised Tasks: Independent of providing a generic 3D representation, our approach to solving the two supervised tasks is novel in a few aspects. There is a large amount of previous work on detecting, describing, and matching image features, either through a handcrafting the feature [5,11,35,38–40] or learning it [9,19,25,49,50,65,66]. Unlike the majority of such features that utilize pre-rectification (within either the method or the training data), we argue that rectification prior to descriptor matching is not required; our representation can learn the impact of viewpoint change, rather than canceling it (by directly training on non-rectified data and supplying camera pose information during training). Therefore, it does not need an apriori rectification and is capable of performing wide baseline matching at the descriptor level. We report state-of-the-art results on feature matching. Wide baseline matching has been also the topic of many papers [24,44,54,62,67] with the majority of them focused on leveraging various geometric constraints for ruling out incorrect 'already-established' correspondences, as well as a number of methods that operate based on generating exhaustive warps [41] or assuming 3D information about the scene is given [61]. In contrast, we learn a descriptor that is supervised to internally handle a wide baseline in the first place.

In the context of pose estimation, we show estimating a 6DOF camera pose given only a pair of local image patches, and without the need for several point correspondences, is feasible. This is different from many previous works [3,8,14,20,26,52,59] from both visual odometery and SfM literature that perform the estimation through a two step process consisting of finding point correspondences between images followed by pose estimation. Koser and Koch [30] also demonstrate pose estimation from a local region, though the plane on which the region lies is assumed to be given. The recent works of [4,29] supervise a ConvNet on the camera pose from image batches but do not provide results on matching and pose estimation. We report a human-level accuracy on this task.

Existing Unsupervised Learning and ConvNet Initialization Works: The majority of previous unsupervised learning, transfer learning, and representation learning works have been targeted towards semantics [17,18,46,47,53]. It has been practically well observed [18,46] that the representation of a convnet trained on imagenet [32] can generalize to other, mostly semantic, tasks. A number of methods investigated initialization techniques for ConvNet training based on unsupervised/weakly supervised data to alleviate the need for a large training dataset for various tasks [17,57]. Very recently, the methods of [4,29] explored using motion metadata associated with videos (KITTI dataset [20]) as a form of supervision for training a ConvNet. However, they either do not investigate developing a 3D representation or intent to provide initialization strategies that are meant to be fine-tuned with supervised data for a desired task. In contrast, we investigate developing a generalizable 3D representation, perform the learning in an object-centric manner, and evaluate its unsupervised performance on various 3D tasks without any fine-tuning on the representation. We experimentally compare against the related recent works that made their models available [4,57].

Primary contributions of this paper are summarized as: (I) A generic 3D representation with empirically validated abstraction and generalization abilities. (II) A learned joint descriptor for wide baseline matching and camera pose estimation at the level of local image patches. (III) A large-scale object-centric dataset of street view scenes including camera pose and correspondence information.

2 Object-Centric Street View Dataset

The dataset for the formulated task needs to not only provide a large amount of training data, but also show a rich camera pose variety, while the scale of the aimed learning problem invalidates any manual procedure. We present a procedure that allows acquiring a large amount of training data in an automated manner, based on two sources of information: (1) Google street view [2] which is an almost inexhaustible source of geo-referenced and calibrated images, (2) 3D city models [1,2] that cover thousands of cities around the world.

The core idea of our approach is to form correspondences between the geo-referenced street view camera and physical 3D points that are given by the 3D models. More specifically, at any given street view location, we densely shoot rays into space in order to find intersections with nearby buildings. Each ray back projects one image pixel into the 3D space, as shown in Fig. 2-(a). By projecting the resulting intersection points onto adjacent street view panoramas (see Fig. 2-b), we can form image to image correspondences (see Fig. 2c). Each image is then associated with a (virtual) camera that fixates on the physical target point on a building by placing it on the optical center. To make the ray intersection procedure scalable, we perform occlusion reasoning on the 3D models to pre-identify from what GPS locations an arbitrary target would be visible and perform the ray intersection on those points only.

Pixel Alignment and Pruning: This system requires integration of multiple resources, including elevation maps, GPS from street view, and 3D models.

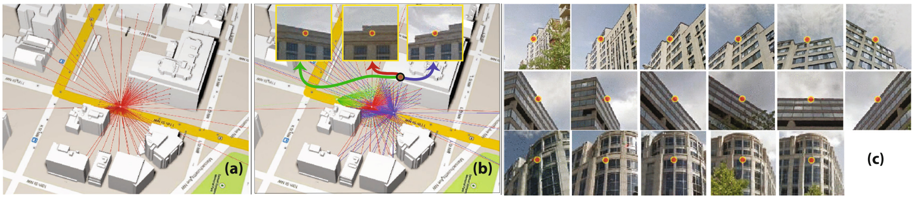

Fig. 2. Illustration of the object-centric data collection process. We use large-scale geo-registered 3D building models to register pixels in street view images on world coordinates system (see (a)) and use that for finding correspondences and their relative pose across multiple street view images (see (b)). Each ray represents one pixel-3D world coordinate correspondence. Each of the red, green, and blue colors represent one street view location. Each row in (c) shows a sample collected image bundle. The center pixel (marker) is expected to correspond to the same physical point. (Color figure online)

Though the quality of output exceeded our expectation (see samples in Fig. 2(c)), any slight inaccuracy in the metadata or 3D models can cause a pixel misalignment in the collected images (examples shown in the first and last rows of Fig. 2(c)). Also, there are undocumented objects such as trees or moving objects that cause occlusions. Thus, a content-based post alignment and pruning was necessary. We again used metadata in our alignment procedure to be able to handle image bundles with arbitrarily wide baselines (note that the collected image bundles can show large, often >100°, viewpoint changes). In the interest of space, we describe this procedure in supplementary material (Sect. 3).

This process forms our dataset composed of matching and non-matching patches as well as the relative camera pose for the matching pairs. We stopped collecting data when we reached the coverage of >200 km^2 from the 7 cities mentioned in Sect. 1. The collection procedure is currently done on Google street view, but can be performed using any geo-referenced calibrated imagery. We will experimentally show that the trained representation on this data does not manifest a clear bias towards street view scenes and outperforms existing feature learning methods on non-street view benchmarks.

Noise Statistics: We performed a user study through Amazon Mechanical Turk to quantify the amount of noise in the final dataset. Please see supplementary material (Sect. 3.2) for the complete discussion and results. Briefly, 68 % of the patch pairs were found to have at least 25 % of overlap in their content. The mean and standard deviation of pixel misalignment was 16.12 (≈11 % of patch width) and 11.55 pixels, respectively. We did not perform any filtering or geo-fencing on top of the collected data as the amount of noise appeared to be within the robustness tolerance of ConvNet trainings and they converged.

3 Learning Using ConvNets

A joint feature descriptor was learnt by supervising a Convolutional Neural Network (ConvNet) to perform 6DOF camera pose estimation and wide baseline matching between pairs of image patches. For the purpose of training, any two image patches depicting the same physical target point in the street view dataset were labelled as matching and other pairs of images were labelled as non-matching. The training for camera pose estimation was performed using matching patches. The patches were always cropped from the center of the collected street view image to keep the optical center at the target point.

The camera pose between each pair of matching patches was represented by a 6D vector; the first three dimensions were Tait-Bryan angles (roll, yaw, pitch) and the last three dimensions were cartesian (x, y, z) translation coordinates expressed in meters. For the purpose of training, 6D pose vectors were preprocessed to be zero mean and unit standard deviation (i.e., z-scoring). The ground-truth and predicted pose vectors for the i^{th} example are denoted by p_i^*, p_i respectively. The pose estimation loss $L_{pose}(p_i^*, p_i)$ was set to be the robust regression loss described in Eq. 1:

$$L_{pose}(p_i^*, p_i) = \begin{cases} e & \text{if } e \leq 1 \\ 1 + \log e & \text{if } e > 1 \end{cases} \quad \text{where } e = ||p_i^* - p_i||_{l_2}. \tag{1}$$

The loss function for patch matching $L_{match}(m_i^*, m_i)$ was set to be sigmoid cross entropy, where m_i^* is the ground-truth binary variable indicating matching/non-matching and m_i is the predicted probability of matching.

ConvNet training was performed to optimize the joint matching and pose estimation loss (L_{joint}) described in Eq. 2. The relative weighting between the pose (L_{pose}) and matching (L_{match}) losses was controlled by λ (we set $\lambda = 1$).

$$L_{joint}(p_i^*, m_i^*, p_i, m_i) = L_{pose}(p_i^*, p_i) + \lambda L_{match}(m_i, m_i^*). \tag{2}$$

Our training set consisted of patch pairs drawn from a wide distribution of baseline changes ranging from 0° to over 120°. We consider patches of size 192 × 192 (<15 % of the actual image size) and rescaled them to 101 × 101 before passing them into the ConvNet.

A ConvNet model with siamese architecture [15] containing two identical streams with identical set of weights was used for computing the relative pose and the matching score between the two input patches. A standard ConvNet architecture was used for each stream: C(20, 7, 1)-ReLU-P(2, 2)-C(40, 5, 1)-ReLU-P(2, 2)-C(80, 4, 1)-ReLU-P(2, 2)-C(160, 4, 2)-ReLU-P(2, 2)-F(500)-ReLU-F(500)-ReLU. The naming convention is as follows: C(n, k, s): convolutional layer n filters, spatial size $k \times k$, and stride s. P(k, s): max pooling layer of size $k \times k$ and stride s. ReLU: rectified linear unit. F(n): fully connected linear layer with n output units. The feature descriptors of both streams were concatenated and fed into a fully connected layer of 500 units which were then fed into the pose and matching losses. With this ConvNet configuration, the size of the image representation (i.e., the last FC vector of one siamese half - see Fig. 1) is 500. Our architecture is admittedly pretty common and standard. This allows us to evaluate if our good end performance is attributed to our hypothesis on learning on foundational tasks and the new dataset, rather than a novel architecture.

We trained the ConvNet model from scratch (i.e., randomly initialized weights) using SGD with momentum (initial learning rate of .001 divided by 10 per 60 K iterations), gradient clipping, and a batch size of 256. We found that the use of gradient clipping was essential for training as even robust regression losses produce unstable gradients at the starting of training. Our network converged after 210 K iterations. Training using Euler angles performed better than quaternions (17.7° vs 29.8° median angular error), and the robust loss outperformed the non-robust l_2 loss (17.7° vs 22.3° median angular error). Additional details about the training procedure can be found in the supplementary material.

4 Experimental Discussions and Results

We implemented our framework using data parallelism [31] on a cluster of 5–10 GPUs. At the test time, computing the representation is a feed-forward pass through a siamese half ConvNet and takes ∼2.9 ms per image on a single processor. Sections 4.1 and 4.2 provide the evaluations of the learned representation on the supervised and novel 3D tasks, respectively.

4.1 Evaluations on the Supervised Tasks

Evaluations on the Street View Dataset. The test set of pose estimation is composed of 7725 pairs of matching patches from our dataset. The test set of matching includes 4223 matching and 18648 non-matching pairs. It is made sure that no data from those areas and their vicinity is used in training. Each patch pair in the test sets was verified by three Amazon Mechanical Turkers to verify the ground truth is indeed correct. For the matching pairs, the Turkers also ensured the center pixel of patches are no more than 25 pixels (∼3 % of image width) apart. Visualizations of the test set can be seen on our website.

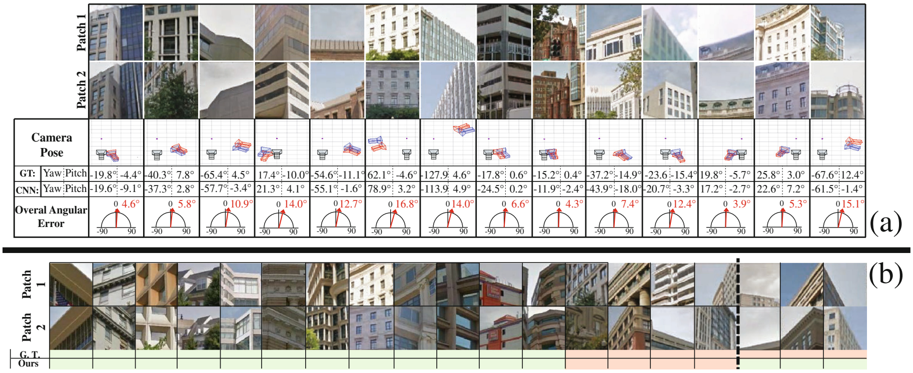
Patch 1
Patch 2
Camera Pose
GT: Yaw Pitch -19.8° -4.4° -40.3° 7.8° -65.4° 4.5° 17.4° -10.0° -54.6° -11.1° 62.1° -4.6° -127.9 4.6° -17.8° 0.6° -15.2° 0.4° -37.2° -14.9° -23.6 -15.4° 19.8° -5.7° 25.8° 3.0° -67.6° 12.4°
CNN: Yaw Pitch -19.6° -9.1° -37.3° 2.8° -57.7° -3.4° 21.3° 4.1° -55.1° -1.6° 78.9° 3.2° -113.9 4.9° -24.5° 0.2° -11.9° -2.4° -43.9° -18.0° -20.7° -3.3° 17.2° -2.7° 22.6° 7.2° -61.5° -1.4°
Overal Angular Error
4.6° 5.8° 10.9° 14.0° 12.7° 16.8° 14.0° 6.6° 4.3° 7.4° 12.4° 3.9° 5.3° 15.1°
0
-90
90
(a)
(b)
Patch 1
Patch 2
G.T.
Ours

Fig. 3. **(a) Sample qualitative results of camera pose estimation**. 1^{st} and 2^{nd} rows show the patches. The 3^{rd} row depicts the estimated relative camera poses on a unit sphere (black: patch 1's camera (reference), red: ground-truth pose of patch 2, blue: estimated pose of patch 2). Rightward and upward are the positive directions. **(b)Sample wide baseline matching results.** Green and red represent 'matching' and 'non-matching', respectively. Three failure cases are shown on the right. (Color figure online)

Pose Estimation. Figure 3-(a) provides qualitative results of pose estimation. The angular evaluation metric is the standard overall angular error [20, 33], defined as the angle between the predicted pose vector and the ground truth vector in the plane defined by their cross product. The translational error metric is l_2 norm of the difference vector between the normalized predicted translation vector and ground truth [20, 33]. The translation vector was normalized to enable comparing with up-to-scale SfM.

Figure 4-right provides the quantitative evaluations. The plots (a) and (c) illustrate the distribution of the test set with respect to pose estimation error for each method (the more skewed to the left, the better). The green curve shows pose estimation results by human subjects. Two users with computer vision

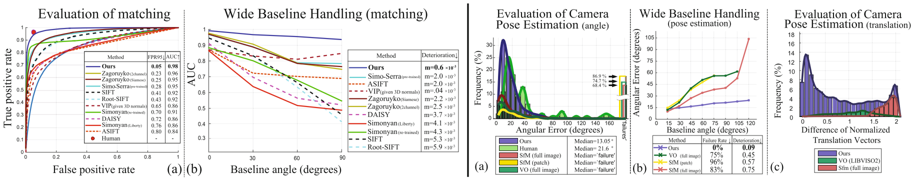

Fig. 4. **Left: Quantitative evaluation of matching.** ROC curves of each method and corresponding AUC and FPR@95 values are shown in (a). **Right: Quantitative evaluation of camera pose estimation.** VO and SfM denote Visual Odometery (LIBVISO2) and Structure-from-Motion (visualSfM), respectively. Evaluation of robustness to wide baseline camera shifts is shown in (b) plots. (Color figure online)

knowledge, but unaware of the particular use case, were asked to estimated the relative pitch and yaw between a random subset of 500 test pairs. They were allowed to train themselves with as many training sampled as they wished. ConvNet outperformed human on this task with a margin of 8° in median error.

Pose Estimation Baselines: We compared against Structure-from-Motion (visualSfM [59,60] with default components and tuned hyper-parameters for pairwise pose estimation on 192 × 192 patches and full images) and LIBVISO2 Visual Odometery [21] on full images. Both SfM and LIBVISO2 VO suffer from a large RANSAC failure rate mostly due to the wide baselines in test pairs.

Figure 4-right (b) shows how the median angular error (Y axis) changes as the baseline of the test pairs (X axis) increases. This is achieved through binning the test set into 8 bins based on their baseline size. This plot quantifies the ability of the evaluated methods in handling a wide baseline. We adopt the slope of the curves as the quantification of deterioration in accuracy as the baseline increases.

Wide Baseline Matching. Figure 3-(b) shows samples feature matching results using our approach, with three failure cases on the right. Figure 4-left provides the quantitative results. The standard metric [12] for descriptor matching is ROC curve acquired from sorting the test set pairs according to their matching score. For unsupervised methods, e.g., SIFT, the matching score is the l_2 distance. False Positive Rate at 95 % recall (FPR@95) and Area Under Curve (AUC) of ROC are standard scalar quantifications of descriptor matching [12,50].

Matching Baselines: We compared our results with the handcrafted features of SIFT [35], Root-SIFT [7], DAISY [55], VIP [61] (which requires the surface normals in the input for which we used the normals from the 3D models), and ASIFT [41]. The matching score of ASIFT was the number of found correspondences in the test pair given the *full images*. We also compared against the learning based features of Zagoruyko and Komodakis [65] (using the models of authors), Simonyan et al. [50] (with and without retraining), Simo-Serra et al. [49] (using authors' best pretrained model) as well as human subjects (the red dot on the

Table 1. Evaluations on Brown's Benchmark [12]. FPR@95 (↓) is the metric.

Train	Test	MatchNet [25]	Zagor. siam [65]	Simonyan [50]	Trzcinski [56]	Brown [12]	Root-SIFT [7]	Ours
Yos	ND	7.70	5.75	6.82	13.37	11.98	22.06	**4.17**
Yos	Lib	13.02	13.45	14.58	21.03	18.27	29.65	**11.66**
Lib	ND	4.75	4.33	7.22	14.15	N/A	22.06	**1.47**
ND	Lib	8.84	8.77	12.42	18.05	16.85	29.65	**7.39**
Lib	Yos	13.57	14.89	**11.18**	19.63	N/A	26.71	13.78
ND	Yos	11.00	13.23	**10.08**	15.86	13.55	26.71	12.30
mean		9.81	10.07	10.38	17.01	15.16	26.14	**8.46**

Table 2. Evaluation on Mikolajczyk and Schmid's [39]. The metric is mAP(↑).

Transf. magnitude	1	2	3	4	5
SIFT [35]	40.1	28.0	24.3	29.0	17.1
Zagor. [65]	43.2	37.5	29.2	28.0	16.8
Fischer et al. [19]	42.3	33.9	26.1	22.1	14.6
Ours-rectified	46.4	**41.3**	29.5	23.7	17.9
Ours-unrectified	**51.4**	37.8	**34.2**	**30.8**	**20.8**

ROC plot). Figure 4-left(b) provides the evaluations in terms of handling wide baselines, similar to Fig. 4-right(b).

Brown et al. Benchmark and Mikolajczyk's Benchmark. We performed evaluations on the *non-street view* benchmarks of Brown et al. [12] and Mikolajczyk and Schmid [39] to find if (1) if our representation was performing well only on street view scenery, and (2) if wide baseline handling capability was achieved at the expense of lower performance on small baselines (as these benchmarks have a narrower baseline compared to our dataset for the most part). Tables 1 and 2 provide the quantitative results. We include a thorough description of evaluation setup and detailed discussions in the supplementary material (Sect. 2).

Joint Feature Learning. We studied different aspects of joint learning the representation and information sharing among the core supervised tasks. In the interest of space, we provide quantitative results in supplementary material (Sect. 1). The conclusion of the tests was that: First, the problems of wide baseline matching and camera pose estimation have a great deal of shared information. Second, one descriptor can encode both problems with no performance drop.

4.2 Evaluating the 3D Representation on Novel Tasks

The results of evaluating our representation on novel 3D tasks are provided in this section. The tasks as well as the images (e.g., Airship images from ImageNet) used in these evaluations are significantly different from what our representation was trained for (i.e., camera pose estimation and matching on local patches of street view images). The fact that, despite such differences, our representation achieves best results among all unsupervised methods and gets close to supervised methods for each of the tasks empirically validates our hypothesis on learning on foundational tasks (see Sect. 1).

Our ways of evaluating and probing the representation in an unsupervised manner are (1) tSNE [36]: large-scale 2D embedding of the representation.

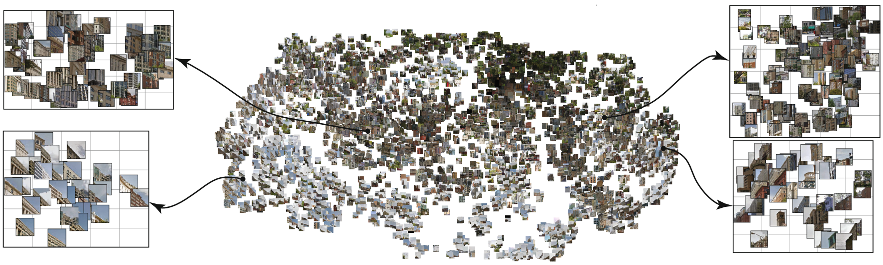

Fig. 5. **2D embedding of our representation on 3,000 unseen patches using tSNE**. An organization based on the Manhattan pose of the patches can be seen. See comparable AlexNet's embedding in the supplementary material's Sect. 6. (best seen on screen)

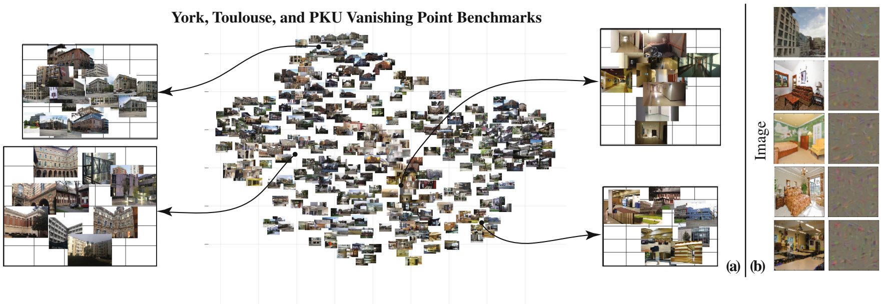

Fig. 6. **(a)** tSNE of a superset of various vanishing point benchmarks [6,16,34] (to battle the small size of datasets). **(b)** inversion [37] of our representation. Both plots shows traits of vanishing points.

This allows visualizing the space and getting a sense of similarity from the perspective of the representation, (2) Nearest Neighbors (NN) on the full dimensional representation, and (3) training a simple classifier (e.g., KNN or a linear classifier) on the *frozen* representation (i.e., no fine-tuning) to read out a desired variable. The latter enables quantifying if the required information for solving a novel task is encoded in the representation and can be extracted using a simple function. We compare against the representations of related methods that made their models available [4,57], various layers of AlexNet trained on ImageNet [32], and a number of supervised techniques for some of the tasks. Additional results are provided in the supplementary material and the website.

Surface Normals and Vanishing Points. Figure 5 shows tSNE embedding of 3,000 unseen patches showing that the organization of the representation space is based on geometry and not semantics/appearance. The ConvNet was trained to estimate the pose between *matching* patches only while in the embedding, the *non-matching* patches with a similar pose are placed nearby. This suggests the representation has generalized the concept of pose to non-matching patches. This indeed has relations to surface normals as the relative pose between an arbitrary

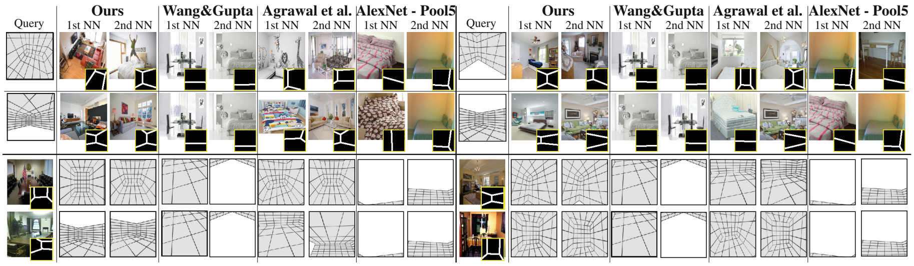

Fig. 7. Scene layout NN search results between LSUN images and synthetic concave cubes defining abstract 3D layouts. Images with yellow boundary show the ground truth layout. (Color figure online)

Table 3. Layout classification (LSUN)

Representation	Classification accuracy
AlexNet FC7	45.9 %
AlexNet Pool5	47.7 %
Ours	57.6%

Table 4. Layout estimation (LSUN)

Method	Corner error	Pixelwise error
UIUC (supervised)	0.11	0.17
Hedau et al. (supervised)	0.15	0.24
Ours (unsupervised)	0.16	0.29

Table 5. Object pose estimation (PASCAL3D)

Method	Av. pose error (°)
scratch	34°
AlexNet (ImaneNet)	23°
Ours	26°

and a frontal patch is equal to the pose of the arbitrary patch; Fig. 5 can be perceived as the organization of the patches based on their surface normals.

To better understand how this was achieved, we visualized the activations of the ConvNet at different layers. Similar to other ConvNets, the first few layers formed general gradient based filters while in higher layers, the edges parallel in the physical world seemed to persist and cluster together. This is similar to the concept of vanishing points, and from the theoretical perspective, would be intriguing and explain the pose estimation results, since three common vanishing points are theoretically enough for a full angular pose estimation [13, 26]. To further investigate this, we generated the inversion of our representation using the method of [37] (see Fig. 6-(b)), which show patterns correlating with the vanishing points of the image. Figure 6-(a) also illustrates the tSNE of a superset of several vanishing point benchmarks showing that images with similar vanishing points are embedded nearby. Therefore, we speculate that the ConvNet has developed a representation based on the concept of vanishing points[2]. This would also explain the results shown in the following sections.

Surface Normal Estimation on NYUv2 [48]**:** Numerical evaluation on unsupervised surface normal estimation provided in supplementary material Sect. 4.

Scene Layout Estimation. We evaluated our representation on LSUN [64] layout benchmark using the standard protocol [64]. Table 4 provides the results

[2] We attempted to quantitatively evaluate this, but the largest vanishing point datasets (e.g., York [16] and PKU [34]) include only 102–200 images for both training and testing. Given a 500D descriptor, it was not feasible to provide a statistically significant evidence.

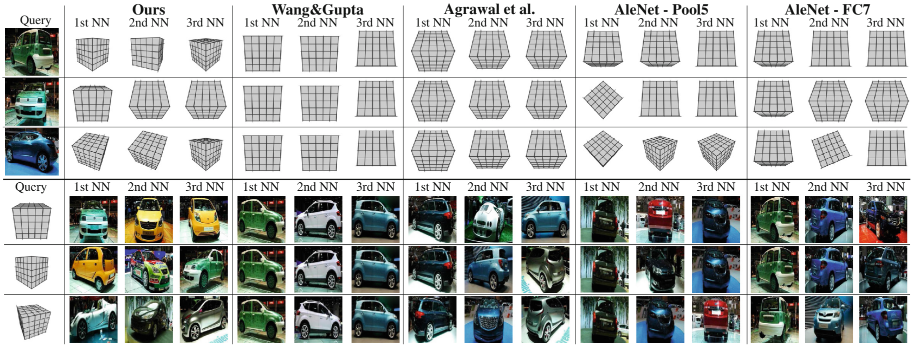

Fig. 8. NN search results between EPFL dataset images and a synthetic cube defining an abstract 3D pose. See the supplementary material (Sect. 5) for tSNE embedding of all cubes and car poses in a joint space. Note that the 3D poses defined by the cubes are 90° congruent.

of layout estimation using a simple NN classifier on our representation along with two supervised baselines, showing that our representation (with no fine-tuning) achieved a performance close to Hedau et al.'s [27] supervised method on this novel task. Table 3 provides the results of layout classification [64] using NN classifier on our representation compared to AlexNets FC7 and Pool5.

Abstraction: Cube⇆Layout: To evaluate the abstract generalization abilities of our representation, we generated a sparse set of 88 images showing the interior of a simple synthetic cube parametrized over different view angles. The rendered images can be seen as an abstract cubic layout of a room. We then performed NN search between these images and LSUN dataset using our representations and several baselines. As apparent in Fig. 7, our representation retrieves meaningful NNs while the baselines mostly overfit to appearance and retrieve either an incorrect or always the same NN. This suggests our representation could abstract away the irrelevant information and encode some information essential to the 3D of the image.

3D Object Pose Estimation.

Abstraction: Cube⇆Object: We performed a similar abstraction test between a set of 88 convex cubes and the images of EPFL Multi-View Car dataset [43], which includes a dense sampling of various viewpoints of cars in an exhibition. We picked this simple cube pattern as it is the simplest geometric element that defines three vanishing points. The same observation as the abstraction experiment on LSUN's is made here with our NNs being meaningful while baselines mostly overfit to appearance with no clear geometric abstraction trait (Fig. 8).

ImageNet: Figure 9 shows the tSNE embedding of several ImageNet categories based on our representation and the baselines. The embeddings of our representation are geometrically meaningful, while the baselines either perform a semantic organization or overfit to other aspects, such as color.

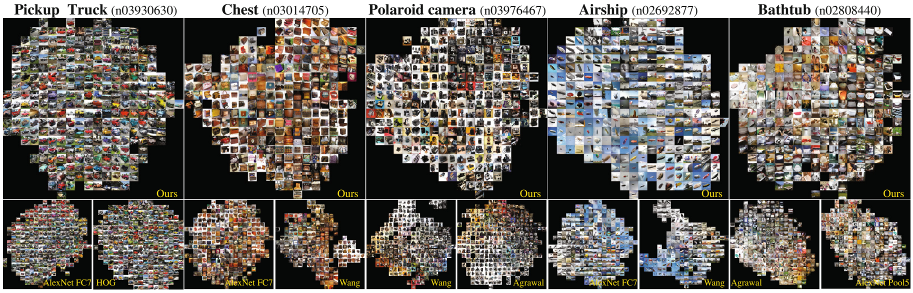

Fig. 9. tSNE of several ImageNet categories using our unsupervised representation along with several baselines. Our representation manifests a meaningful geometric organization of objects. tSNE of more categories in the supplementary material and the website. (best seen on screen) (Color figure online)

PASCAL3D: Figure 10 shows cross-category NN search results for our representation along with several baselines. This experiment also evaluates a certain level of abstraction as some of the object categories can be drastically different looking. We also quantitatively evaluated on 3D object pose estimation on PASCAL3D. For this experiment, we trained a ConvNet from scratch, fine-tuned AlexNet pre-trained on ImageNet, and fine-tuned our network; we read the pose out using a linear regressor layer.[3] Our results outperform scratch network and come close to AlexNet that has seen thousands of images from the same categories from ImageNet and other objects (Table 5). Note that certain aspects of object pose estimation, e.g., distinguishing between the front and back of a bus, are more of a semantic task rather than geometric/3D. This explains a considerable part of the failures of our representation which is object/semantic agnostic.

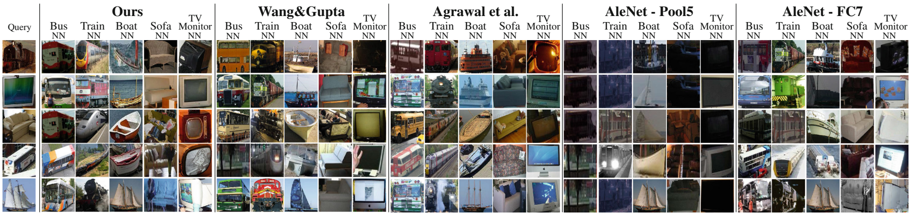

Fig. 10. Qualitative results of cross-category NN-search on PASCAL3D using our representation along with baselines.

[3] The classes of boat, sofa, and chair were showing a performance near statistical random for all methods and were removed from the evaluations.

5 Discussion and Conclusion

To summarize, we developed a generic 3D representation through solving a set of supervised foundational proxy tasks. We reported state-of-the-art results on the supervised tasks and showed the learned representation manifests generalization and abstraction traits. However, a number of questions remain open:

Though we were inspired by cognitive studies in defining the foundational supervised tasks leading to a generalizable representation, this remains at an inspiration level. Given that a 'taxonomy' among basic 3D tasks has not been developed, it is not concretely defined which tasks are foundational and which ones are secondary. Developing such a taxonomy (i.e., whether task A is inclusive of, overlapping with, or disjoint from task B) or generally efforts understanding the task space would be a rewarding step towards soundly developing the *3D complete* representation. Also, semantic and 3D aspects of the visual world are tangled together. So far, we have developed independent semantic and 3D representations, but investigating concrete techniques for integrating them (beyond simplistic late fusion or ConvNet fine-tuning) is a worthwhile future direction for research. Perhaps, inspirations from partitions of visual cortex could be insightful towards developing the ultimate *vision complete* representation.

Acknowledgement. MURI (1186514-1-TBCJE), Nissan (1188371-1-UDARQ).

References

1. http://opendata.dc.gov/
2. Google Street View. https://www.google.com/maps/streetview/
3. Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S.M., Szeliski, R.: Building Rome in a day. Commun. ACM **54**(10), 105–112 (2011)
4. Agrawal, P., Carreira, J., Malik, J.: Learning to see by moving (2015)
5. Alahi, A., Ortiz, R., Vandergheynst, P.: FREAK: fast retina keypoint. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 510–517. IEEE (2012)
6. Angladon, V., Gasparini, S., Charvillat, V.: The toulouse vanishing points dataset. In: Proceedings of the 6th ACM Multimedia Systems Conference (MMSys 2015) (2015)
7. Arandjelovic, R., Zisserman, A.: Three things everyone should know to improve object retrieval. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2911–2918. IEEE (2012)
8. Badino, H., Yamamoto, A., Kanade, T.: Visual odometry by multi-frame feature integration. In: 2013 IEEE International Conference on Computer Vision Workshops (ICCVW), pp. 222–229. IEEE (2013)
9. Balntas, V., Johns, E., Tang, L., Mikolajczyk, K.: PN-Net: conjoined triple deep network for learning local image descriptors. arXiv preprint arXiv:1601.05030 (2016)
10. Banks, M.S., Salapatek, P.: Infant visual perception. In: Mussen, P.H. (eds.) Handbook of Child Psychology: Formerly Carmichael's Manual of Child Psychology (1983)

11. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)
12. Brown, M., Hua, G., Winder, S.: Discriminative learning of local image descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **33**(1), 43–57 (2011)
13. Caprile, B., Torre, V.: Using vanishing points for camera calibration. Int. J. Comput. Vis. **4**(2), 127–139 (1990)
14. Chen, D.M., Baatz, G., Köser, K., Tsai, S.S., Vedantham, R., Pylvä, T., Roimela, K., Chen, X., Bach, J., Pollefeys, M., et al.: City-scale landmark identification on mobile devices. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 737–744. IEEE (2011)
15. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 539–546. IEEE (2005)
16. Denis, P., Elder, J.H., Estrada, F.J.: Efficient edge-based methods for estimating manhattan frames in urban imagery. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 197–210. Springer, Heidelberg (2008)
17. Doersch, C., Gupta, A., Efros, A.A.: Unsupervised visual representation learning by context prediction. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1422–1430 (2015)
18. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: DeCAF: a deep convolutional activation feature for generic visual recognition. arXiv preprint arXiv:1310.1531 (2013)
19. Fischer, P., Dosovitskiy, A., Brox, T.: Descriptor matching with convolutional neural networks: a comparison to SIFT (2014). arXiv preprint arXiv:1405.5769
20. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3354–3361. IEEE (2012)
21. Geiger, A., Ziegler, J., Stiller, C.: StereoScan: dense 3d reconstruction in real-time. In: Intelligent Vehicles Symposium (IV) (2011)
22. Gibson, E.J., Walk, R.D.: The Visual Cliff, vol. 1. WH Freeman Company, New York (1960)
23. Girshick, R.: Fast R-CNN. In: Proceedings of the International Conference on Computer Vision (ICCV) (2015)
24. Goedemé, T., Tuytelaars, T., Van Gool, L.: Fast wide baseline matching for visual navigation. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. I–24 (2004)
25. Han, X., Leung, T., Jia, Y., Sukthankar, R., Berg, A.C.: MatchNet: unifying feature and metric learning for patch-based matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3279–3286 (2015)
26. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge (2003)
27. Hedau, V., Hoiem, D., Forsyth, D.: Recovering the spatial layout of cluttered rooms. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 1849–1856. IEEE (2009)
28. Held, R., Hein, A.: Movement-produced stimulation in the development of visually guided behavior. J. Comp. Physiol. Psychol. **56**(5), 872 (1963)

29. Jayaraman, D., Grauman, K.: Learning image representations tied to ego-motion. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1413–1421 (2015)
30. Köser, K., Koch, R.: Differential spatial resection - pose estimation using a single local image feature. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 312–325. Springer, Heidelberg (2008)
31. Krizhevsky, A.: One weird trick for parallelizing convolutional neural networks. arXiv preprint arXiv:1404.5997 (2014)
32. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
33. Kümmerle, R., Steder, B., Dornhege, C., Ruhnke, M., Grisetti, G., Stachniss, C., Kleiner, A.: On measuring the accuracy of SLAM algorithms. Auton. Robot. **27**(4), 387–407 (2009)
34. Li, B., Peng, K., Ying, X., Zha, H.: Simultaneous vanishing point detection and camera calibration from single images. In: Boyle, R., et al. (eds.) ISVC 2010, Part II. LNCS, vol. 6454, pp. 151–160. Springer, Heidelberg (2010)
35. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
36. Van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(2579–2605), 85 (2008)
37. Mahendran, A., Vedaldi, A.: Understanding deep image representations by inverting them. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5188–5196. IEEE (2015)
38. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. Image Vis. Comput. **22**(10), 761–767 (2004)
39. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **27**(10), 1615–1630 (2005)
40. Moreels, P., Perona, P.: Evaluation of features detectors and descriptors based on 3D objects. Int. J. Comput. Vis. **73**(3), 263–284 (2007)
41. Morel, J.M., Yu, G.: ASIFT: a new framework for fully affine invariant image comparison. SIAM J. Imaging Sci. **2**(2), 438–469 (2009)
42. Nistér, D., Naroditsky, O., Bergen, J.: Visual odometry. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, vol. 1, pp. I–652. IEEE (2004)
43. Ozuysal, M., Lepetit, V., Fua, P.: Pose estimation for category specific multiview object localization. In: Conference on Computer Vision and Pattern Recognition, Miami, FL, June 2009
44. Pritchett, P., Zisserman, A.: Wide baseline stereo matching. In: Sixth International Conference on Computer Vision, 1998, pp. 754–760. IEEE (1998)
45. Rader, N., Bausano, M., Richards, J.E.: On the nature of the visual-cliff-avoidance response in human infants. Child Dev. 61–68 (1980)
46. Razavian, A., Azizpour, H., Sullivan, J., Carlsson, S.: CNN features off-the-shelf: an astounding baseline for recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 806–813 (2014)
47. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: OverFeat: integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229 (2013)

48. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from RGBD images. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part V. LNCS, vol. 7576, pp. 746–760. Springer, Heidelberg (2012)
49. Simo-Serra, E., Trulls, E., Ferraz, L., Kokkinos, I., Fua, P., Moreno-Noguer, F.: Discriminative learning of deep convolutional feature point descriptors. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 118–126 (2015)
50. Simonyan, K., Vedaldi, A., Zisserman, A.: Learning local feature descriptors using convex optimisation. IEEE Trans. Pattern Anal. Mach. Intell. **36**(8) (2014)
51. Smith, L., Gasser, M.: The development of embodied cognition: six lessons from babies. Artif. Life **11**(1–2), 13–29 (2005)
52. Song, S., Chandraker, M., Guest, C.C.: Parallel, real-time monocular visual odometry. In: 2013 IEEE International Conference on Robotics and Automation (ICRA). IEEE (2013)
53. Tarr, M.J., Black, M.J.: A computational and evolutionary perspective on the role of representation in vision. CVGIP: Image Underst. **60**(1), 65–73 (1994)
54. Tell, D., Carlsson, S.: Combining appearance and topology for wide baseline matching. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part I. LNCS, vol. 2350, pp. 68–81. Springer, Heidelberg (2002)
55. Tola, E., Lepetit, V., Fua, P.: A fast local descriptor for dense matching. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8. IEEE (2008)
56. Trzcinski, T., Christoudias, M., Lepetit, V., Fua, P.: Learning image descriptors with the boosting-trick. In: Advances in Neural Information Processing Systems, pp. 269–277 (2012)
57. Wang, X., Gupta, A.: Unsupervised learning of visual representations using videos. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2794–2802 (2015)
58. Weston, J., Ratle, F., Mobahi, H., Collobert, R.: Deep learning via semi-supervised embedding. In: Montavon, G., Orr, G.B., Müller, K.-R. (eds.) Neural Networks: Tricks of the Trade, 2nd edn. LNCS, vol. 7700, 2nd edn, pp. 639–655. Springer, Heidelberg (2012)
59. Wu, C.: VisualSFM: a visual structure from motion system (2011). http://ccwu.me/vsfm/
60. Wu, C., Agarwal, S., Curless, B., Seitz, S.M.: Multicore bundle adjustment. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3057–3064. IEEE (2011)
61. Wu, C., Clipp, B., Li, X., Frahm, J.M., Pollefeys, M.: 3D model matching with viewpoint-invariant patches (VIP). In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8. IEEE (2008)
62. Xiao, J., Shah, M.: Two-frame wide baseline matching. In: Proceedings of the Ninth IEEE International Conference on Computer Vision, 2003, pp. 603–609. IEEE (2003)
63. Xu, C., Lu, C., Liang, X., Gao, J., Zheng, W., Wang, T., Yan, S.: Multi-loss regularized deep neural network. IEEE Trans. Circuits Syst. Video Technol. **PP**(99), 1–1 (2015)

64. Yu, F., Zhang, Y., Song, S., Seff, A., Xiao, J.: Construction of a large-scale image dataset using deep learning with humans in the loop. arXiv preprint arXiv:1506.03365 (2015)
65. Zagoruyko, S., Komodakis, N.: Learning to compare image patches via convolutional neural networks (2015). arXiv preprint arXiv:1504.03641v1
66. Zbontar, J., LeCun, Y.: Computing the stereo matching cost with a convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1592–1599 (2015)
67. Zhang, Z., Ganesh, A., Liang, X., Ma, Y.: TILT: transform invariant low-rank textures. Int. J. Comput. Vis. **99**(1), 1–24 (2012)