



**HAL**  
open science

# Textured Object Recognition: Balancing Model Robustness and Complexity

Guido Manfredi, Michel Devy, Daniel Sidobre

► **To cite this version:**

Guido Manfredi, Michel Devy, Daniel Sidobre. Textured Object Recognition: Balancing Model Robustness and Complexity. 16th International Conference on Computer Analysis of Images and Patterns (CAIP 2015), Sep 2015, La Valette, Malta. 10.1007/978-3-319-23192-1\_5 . hal-01355103

**HAL Id: hal-01355103**

**<https://hal.science/hal-01355103>**

Submitted on 22 Aug 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Textured Object Recognition: Balancing Model Robustness and Complexity

Guido Manfredi<sup>1,2</sup>, Michel Devy<sup>2</sup>, and Daniel Sidobre<sup>1,2</sup>

<sup>1</sup> LAAS CNRS, F-31400 Toulouse, France

<sup>2</sup> Univ. de Toulouse, UPS, Toulouse, France  
`gmanfred, michel, daniel@laas.fr`,

**Abstract.** When it comes to textured object modelling, the standard practice is to use a multiple view approach. The numerous views allow reconstruction and provide robustness to viewpoint change but yield complex models. This paper shows that robustness with lighter models can be achieved through robust descriptors. A comparison between various descriptors allows choosing the one providing the best viewpoint robustness, in this case the ASIFT descriptor. Then, using this descriptor, the results show, for a wide variety of object shapes, that as few as seventeen views provide a high level of robustness to viewpoint change while being fast to process and using small memory space. This work concludes advocating in favour of modelling methods using robust descriptors and a small number of views.

**Keywords:** object modelling, object recognition, multiple views, robust descriptors

## 1 Introduction

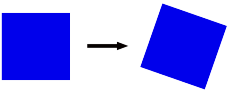
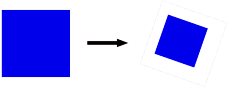
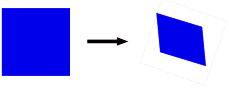
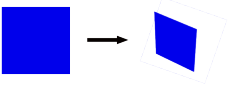
For textured objects recognition, current modelling methods rely on hundreds of views to build models containing tens of thousands of local descriptors.

In object recognition problems, an image is compared to known models to identify objects present in the scene. The objects of interest can have any pose, i.e. the object-to-image transform can be of any type, see the transform types Table 1. To ensure recognition under any transform, current modelling methods aim for viewpoint invariant models. This is generally achieved by including numerous views of the object and produces complex models. The drawback of such approaches is that complex models are harder to build, slower to process and their many descriptors increase the chances of confusion.

We believe that, for many applications, full invariance is overkill and some blind spots in the model are acceptable. Moreover, this paper advocates in favour of using robust descriptors to reduce the number of views required to achieve a given level of viewpoint robustness.

Indeed, robustness to viewpoint change can be achieved through two different paradigms.

Table 1: List of transforms that the image of an object can undergo depending on the viewpoint. The variables  $t_x$ ,  $t_y$ ,  $r_{i,j}$ ,  $s$ ,  $a_{i,j}$  and  $h_{i,j}$  are respectively translation, rotation, scale, affine and homography coefficients.

Name	Matrix	Figures
Euclidean	$\begin{pmatrix} r_{1,1} & r_{1,2} & t_x \\ r_{2,1} & r_{2,2} & t_y \\ 0 & 0 & 1 \end{pmatrix}$	
Similarity	$\begin{pmatrix} sr_{1,1} & sr_{1,2} & t_x \\ sr_{2,1} & sr_{2,2} & t_y \\ 0 & 0 & 1 \end{pmatrix}$	
Affine	$\begin{pmatrix} a_{1,1} & a_{1,2} & t_x \\ a_{2,1} & a_{2,2} & t_y \\ 0 & 0 & 1 \end{pmatrix}$	
Projective	$\begin{pmatrix} h_{1,1} & h_{1,2} & h_{1,3} \\ h_{2,1} & h_{2,2} & h_{2,3} \\ h_{3,1} & h_{3,2} & h_{3,3} \end{pmatrix}$	

- Multiple Views: The model can be made robust to a given transform type by including multiple views of the object, under a wide variety of the given transform. For example, to achieve in-plane rotation invariance, the model should include multiple images of the object with different in-plane rotation angles. This method requires numerous views and creates a complex model.
- Robust Descriptors: Some image descriptors already carry their own invariance to given transforms. By using these descriptors, the resulting model acquires the descriptor’s invariance. For example, Harris cornerness descriptors [5] are invariant to in-plane rotation. Thus, a model using Harris descriptors is invariant to object rotation in the camera plane. Usually, a descriptor robust to complex transforms has higher dimensionality.

Though the multiple views paradigm is the most popular, this paper shows that a correct balance between both paradigms produces light models with high robustness to viewpoint change.

The contribution of this work is twofold. First, a methodology to assess the robustness of a recognition model is proposed in Section 3. Scanning an object observation half-sphere allows finding the blind spots positions and size. Then, this methodology is put to use on 272 models to get an insight into the balance between number of views and viewpoint robustness. Results presented in Section 4 suggest that as few as seventeen views provide high robustness while keeping complexity low.

The next section describes the multiple view and robust descriptors paradigms in more detail and covers the corresponding literature.

## 2 Previous Works

Few works have tackled the specific problem of modelling for recognition [15, 12]. In a notable effort, Waibel et al. [21] created a tool to build models by scanning objects. But it is hard to control the complexity of the resulting models which tend to be large and complex. This section present the two main paradigms used when modelling for recognition and their state of the art.

### 2.1 Multiple Views Paradigm

The general idea is to build an object's 3-D model, from pictures or a video stream. Then, synthetic views of the 3-D model are generated by moving a virtual camera around it under various transforms. Descriptors are extracted from each view and added to the model. More different views imply more robustness to viewpoint change. The aim is to have numerous views for all type of image-to-object transform. To the best of our knowledge, few works tried to merge the 3-D modelling and recognition modelling processes [15].

To build the 3-D model, the object recognition community relies on techniques originating both from photogrammetry [20] and Structure from Motion [13]. These methods allow building a 3-D model of the observed scene from a large set of overlapping images. By matching point features on two images and using multiple view geometry [6], one can compute the motion between the two cameras. The motion information allows triangulating [7] the matched features to get 3-D points, i.e. the structure information. The idea can be applied offline or online.

The offline methods use a fixed set of images, or views, to create a model. Such algorithm selects two initial images to compute a baseline, the first motion, according to some criterion. This is often done by adding a known pattern to the scene in order to set the scale factor. Then, the other images are added to the model one by one. Finally, a bundle adjustment algorithm is run on the model. As an example, the software Bundler [18] allows accurate reconstruction of large scenes, like a city [19], in the form of a point cloud. It can be combined with the work from Ponce and Furukawa [4] which allows turning this point cloud into a full 3-D model. The main drawback is that the views must be taken carefully to have enough overlap and cover the scene properly. Recently, Autocad made freely available a software, 123DCatch [2], which allows creating a textured model of an object from a limited set of views. The modelling itself is made offline, in the cloud. Then, the user retrieves a textured 3-D model. To facilitate access for non experts, the software guides the user in finding good views for the model. But even with this help, getting a precise 3-D model requires expertise and understanding of the underlying techniques.

The online methods build a model incrementally by adding new views as they are taken, usually from a video stream. The overlap between views is computed automatically and a new view is added when the overlap is not sufficient for precise motion estimation. Usually, while modelling is under way, a bundle adjustment algorithm enforces coherence within the data. The work from [12] shows an online object modelling program using the 3-D model under construction to help further modelling. They also regularly run a bundle adjustment for increased precision. Royer et al. [16] proposed a similar method for modelling but the bundle adjustment is only run on the most recent parts of the reconstructed model.

All these methods share the same defect, they need numerous images to cover all possible transforms. This yields large and complex models.

## 2.2 Robust Descriptors Paradigm

Instead of capturing views of the object under different transforms, the robust descriptors paradigm relies on descriptors robust to these transforms. As the descriptors handle the robustness to transforms, a smaller amount of views is required. The following goes over some of the most popular descriptors and sums up their robustness in Table 2.

The corneriness descriptor [5] captures the structure of the local neighbourhood with an auto-correlation matrix. It is robust to translations and rotations. The SIFT descriptor [9] relies on histograms of gradients and a multi-scale simulation to achieve high degree of robustness against translation, rotation and scale. The authors of the SURF descriptor [3] use Haar wavelets to approximate the hessian over a local image patch. The integral image technique allows fast computation. These approaches are robust to translation, rotation and scale. Recently, Yu et al. proposed the ASIFT descriptor [22], an extension of the SIFT descriptor which handles affine transforms. The image being described is simulated under different affine transforms, providing robustness to affine transforms.

These descriptors have a high discriminative power at the cost of a high dimensionality. Inspired by the CENSUS transform [23], various works have tackled this problem by focusing on binary descriptors. The ORB descriptor [17] uses differences of pixels to form a binary vector. Proposed by Alahi et al. and inspired by the retina [1], the FREAK descriptor uses differences of Gaussians to create a binary string.

Finally, the so-called Calonder descriptor [11] is created through a learning process. This descriptor learns, on a set of images, how to be robust against different transforms that are present in the training set. Though having showed good results, it is seldom used. For a comparison of local descriptors performances, see the work from Mikolajczyk et al. [10].

This section has presented the two main paradigms available when modelling an object. They are not exclusive, but finding a right balance between the number of views and strength of features while keeping the model complexity low is a hard task.

Table 2: List of some classical descriptors along with their respective invariances.

Name	Rotation	Scale	Affine
Harris	+	-	-
SURF	+	+	-
SIFT	+	+	-
ASIFT	+	+	+
ORB	+	+	-
FREAK	+	+	-
Calonder	Depends	on	training

In the next chapter a robust descriptor is chosen for our experiments. Then various models are compared, with different number of views, to provide an insight about how to find a right balance between robustness to viewpoint change, model complexity and processing time.

### 3 Comparing Models

The general idea is to build a model with a given number of views and a descriptor type. Then scan the observation half-sphere around the object and find out from how many points of view the object is correctly recognized. Scanning the full observation half-sphere in a continuous way is not possible, so it is sampled regularly as illustrated in Figure 1. This sampling method tend to over-represent the sphere’s poles. In this work, care is taken to keep some distance between the sampling points and the pole. In the experiments, the minimum theta angle is 30.

#### 3.1 Comparison Method

Consider a set of RGB views of an object, these views form the object set. To create a model, one selects  $D$ , a descriptor type, and  $N$  views from the object set. These  $N$  views form the model set. The model set is for training and the object set is for testing.

Each view in the object set is compared with each view of the model set, using the descriptors  $D$ . For each couple of object-model views, matches are extracted and filtered using the fundamental matrix.

For a given object view, the best matching model view is the one with the maximum number of remaining matches. A percentage is obtained by dividing the number of matches by the total number of descriptors in the object view. If the percentage of matches is superior to a threshold, fixed for the whole experiment, then the recognition is considered as successful for this pose.

#### 3.2 Comparison metrics

When building a model for recognition, three properties are desirable: robustness to viewpoint changes, low size and low processing time.

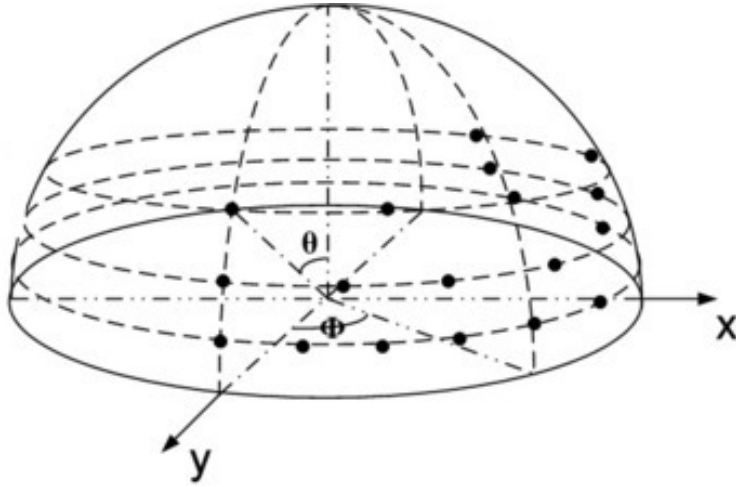


Fig. 1: The observation half-sphere around an object sampled with angular step  $\theta$  and  $\phi$ .

Viewpoint robustness is measured by quantifying the size of the model's blind regions. The blind regions are the angles from which a camera would not be able to recognize the object, for a given model. The blind region size is measured with the total blind region, i.e. the sum of all blind regions, expressed in degrees. The smaller total blind region, the higher viewpoint robustness.

The size of a model has no direct impact on the recognition performances. However, it is crucial when scaling up the number of objects. Models weighting various gigabytes will be hard to store when becoming numerous. It can even come to the point where disc access speed may slow down the overall process. The model size is measured in megabytes.

Finally, the complexity of a model is crucial for processing speed. It depends on the number of views incorporated in the model, and the dimensionality of the descriptors used to characterise these views. The processing speed is computed in milliseconds.

The next section describes two experiments to help select a robust descriptor and assess the viewpoint robustness, size and processing time of models depending on the number of views used to create them.

## 4 Experiments & Results

Some descriptors provide more robustness to viewpoint change but at the cost of longer processing time and/or higher model complexity. Before comparing models, the dataset is presented and the choice of the ASIFT descriptor for this experiment is justified.

#### 4.1 Dataset: Washington RGB-D

This dataset comes from the work of Lai et al. [8]. It has been acquired by rotating an object on a turntable in front of a couple of sensors equivalent to a RGB-D camera.

For a given object, the camera samples the observation half-sphere with an approximate step of  $9^\circ$  around the vertical axis; and at positions of  $30^\circ$ ,  $45^\circ$  and  $60^\circ$  above the horizontal plan. The resulting dataset contains rgb and depth pictures, though only the rgb pictures are used for this experiment. There are roughly 250 pictures per object, for 300 objects from 51 categories. Because a minimum of texture is necessary for this experiment, only the following categories are considered: cereal box, food bag, food box, food can, food jar, instant noodles, soda can, water bottle. This makes a total of 68 objects from 8 classes, see Figure 2. The goal when using various objects is to consider different objects geometries. In this dataset, objects in the same class have similar geometry. For this reason, the results shown are the mean over the objects of each class.



Fig. 2: An example for each of the considered classes. From left to right: cereal box, food bag, food box, food can, food jar, instant noodles, soda can, water bottle.

#### 4.2 Preliminary experiment

The goal of this preliminary experiment is to select the best suited descriptor for the rest of the experiments. To do so, an object is matched with a simplified model made of two views, chosen arbitrarily at  $90^\circ$  and  $270^\circ$ . Various descriptors are used and the one yielding the highest percent of matches is retained. This is done for five descriptor types: ASIFT, SIFT, SURF, ORB and FREAK. Each descriptor is computed on key points obtained with their classically associated key point detector; respectively SIFT, SIFT, SURF, Oriented FAST and FAST [14] detectors.



Because the model and the object contain two identical views, the  $90^\circ$  and  $270^\circ$  views, two peaks in the percent of matches are expected, corresponding to these identical views being matched, and few matches when moving away from the peaks. Note that the curves represent the mean over all objects of a class. These objects are not perfectly aligned, so the peak can occur at slightly different places for each object. When computing the mean, this tend to create a lower and larger peak which does not reach the 100% matches. Also note that the curves represent the data for three tilt positions,  $15^\circ$ ,  $35^\circ$  and  $60^\circ$ . At best, each peak is subdivided into three successive peaks, one for each tilt position. For clarity, the overall results are commented and illustrated with the soda can and food can classes.

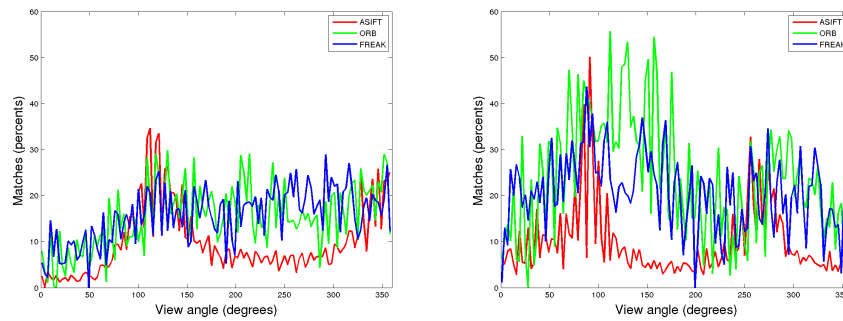


Fig. 3: Results for the soda can and food can classes with ORB and FREAK descriptors for a two views model. ASIFT is provided for comparison.

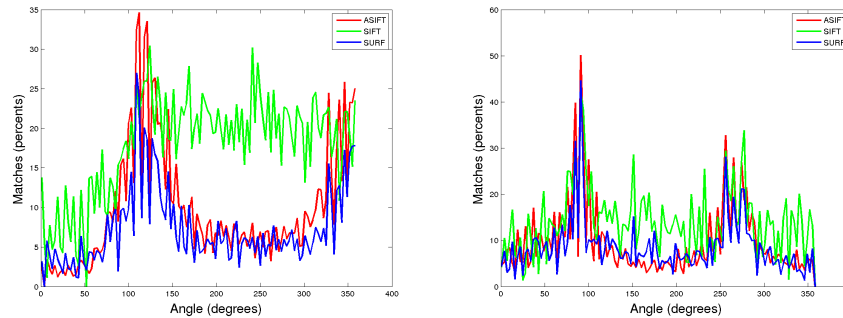


Fig. 4: Results for the soda can and food can classes with ASIFT, SIFT and SURF descriptors for a two views model.

As can be seen in Figure 3, ORB, and FREAK descriptors show numerous peaks and find many matches independent of the angle, thus false matches. Results for other classes are similar. These descriptors have the particularity of being binary descriptors, fast to match but with lower dimensionality, thus lower discriminative power.

Figure 4 shows that SIFT tend to have a high match percentage where there are no training images, thus false matches. ASIFT and SURF descriptors behave similarly showing peaks around the training images positions and low percent of matches elsewhere, but ASIFT tend to score slightly higher. ASIFT uses an affine approximation of the object to account for affine transforms. As noted by Lowe in [9], for a given affine transform, the approximation error can be bounded by the projected diameter of the object’s circumscribing sphere times a constant. It seems reasonable to believe that SURF and ASIFT giving similar results hints to the fact that this affine approximation is not valid for objects of this size. Nevertheless, because of its scores, ASIFT is chosen to perform the experiment.

### 4.3 Models comparison

Given the descriptor type and an object, successive models are created using more and more views. The models are compared with the comparison metrics described in the previous section in order to look for the best compromise between model robustness, model size and processing speed. Using the ASIFT descriptor, four models are built using 4, 8, 17 and 33 views uniformly distributed, with tilt  $15^\circ$ . Each model is then matched to the full object. Two views are considered as matching if 25% of their key points are matching. This threshold is chosen empirically.

As can be seen in Table 3, there is a great inter class variation due to the objects having different shapes and texture size. For the particular case of the food can class, a very simple model with 4 views is sufficient for recognition with few blind spots. This may be due to the cylindrical shape which provide features appearance changing smoothly.

Otherwise, 4 views are not sufficient in terms of size of blind spots. Using 8 views seems to produce few blind spots, except for the food bag and instant noodles classes. The speed and model size remain low. Overall, a model with 17 views provides a good compromise in terms of size of blind spots and size of model. However the matching speed is almost as high as when using 33 views. For a model with 33 views, the blind spots sum is small but the matching time can go up to 10 ms and the size of the model up to 20.5 Mb.

Between a model with 4 and 33 views there can be a difference of up to 28 times more blind spots (food jar class). But the processing time and model size are divided by 10. Using 17 views tend to produce models twice lighter than using 33 views, and the processing time can be smaller or equal. The difference in blind spots sum can be divided up to 4 times.

Table 3: For each considered class, this Table shows the number of views, total sum of blind spots, mean matching time and mean size of the model. The mean and standard deviation over all classes is provided at the bottom row. This experiment considers 3 tilt and 360°pan positions, the total spanned angle is  $360 * 3 = 1080^\circ$

Class	Views	Blind spots(deg)	Time (ms)	Size (Mb)
cereal box	4	630	3.1	2.3
	8	279	4.8	4.8
	17	198	9.9	10.6
	33	108	10.7	20.5
food bag	4	981	2.3	1.7
	8	882	3.6	4.1
	17	639	7.7	8.7
	33	378	6.7	17.1
instant noodles	4	945	0.8	0.42
	8	765	1.7	1.0
	17	432	3.6	2.2
	33	306	7.2	4.3
soda can	4	540	0.7	0.25
	8	198	1.5	0.51
	17	63	3.2	1.1
	33	36	6.4	2.15
food box	4	639	0.9	0.85
	8	306	1.9	2.17
	17	117	4.2	4.6
	33	45	8.1	9.0
food can	4	234	0.7	0.24
	8	0	1.3	0.48
	17	0	3	1.04
	33	0	5.5	2.0
food jar	4	432	0.6	0.24
	8	144	1.2	0.48
	17	45	2.8	1.05
	33	18	5.5	2.04
water bottle	4	873	0.9	0.54
	8	603	1.7	1.1
	17	252	3.7	2.4
	33	63	7.2	4.63
<b>mean / std-dev</b>	<b>4</b>	<b>659/245</b>	<b>1.2/0.8</b>	<b>0.8/0.7</b>
	<b>8</b>	<b>397/295</b>	<b>2.2/1.2</b>	<b>1.8/1.6</b>
	<b>17</b>	<b>218/205</b>	<b>4.8/2.4</b>	<b>4.0/3.5</b>
	<b>33</b>	<b>119/133</b>	<b>4.7/5.1</b>	<b>7.7/6.8</b>

Using 33 views provides a model close to covering the whole observation half-sphere. Still, this is a small number when compared with the hundreds of views contained in models created with Structure from Motion techniques.

## 5 Conclusion

This work shows that a small number of views is sufficient to create models robust to out of plane rotation. Added to the translation invariance, in plane rotation robustness and scale invariance of the considered features it yields models highly robust to viewpoint change. The small image set compensate for the complexity of the descriptors and overall the models are light and fast to process when compared with models using hundreds of views. Though creating light models seems to be the way to go, most modelling methods rely on overlapping images and does not allow to do so. Future work involves developing methods to create light yet robust models in a simple fashion.

## References

1. Alahi, A., Ortiz, R., Vandergheynst, P.: Freak: Fast retina keypoint. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. pp. 510–517. Ieee (2012)
2. Autodesk: 123d catch. <http://www.123dapp.com/catch> (2014), accessed: 2010-09-30
3. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). *Computer vision and image understanding* 110(3), 346–359 (2008)
4. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 32(8), 1362–1376 (2010)
5. Harris, C., Stephens, M.: A combined corner and edge detector. In: *Alvey vision conference*. vol. 15, p. 50. Manchester, UK (1988)
6. Hartley, R., Zisserman, A.: *Multiple view geometry in computer vision*. Cambridge university press (2003)
7. Hartley, R.I., Sturm, P.: Triangulation. *Computer vision and image understanding* 68(2), 146–157 (1997)
8. Lai, K., Bo, L., Ren, X., Fox, D.: A large-scale hierarchical multi-view rgb-d object dataset. In: *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. pp. 1817–1824. IEEE (2011)
9. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *International journal of computer vision* 60(2), 91–110 (2004)
10. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 27(10), 1615–1630 (2005)
11. Ozuysal, M., Calonder, M., Lepetit, V., Fua, P.: Fast keypoint recognition using random ferns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 32(3), 448–461 (2010)
12. Pan, Q., Reitmayr, G., Drummond, T.: Proforma: Probabilistic feature-based online rapid model acquisition. In: *BMVC*. pp. 1–11 (2009)
13. Pollefeys, M., Koch, R., Vergauwen, M., Van Gool, L.: Flexible acquisition of 3d structure from motion. In: *Proc. IEEE workshop on Image and Multidimensional Digital Signal Processing*. Citeseer (1998)

14. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In: *Computer Vision—ECCV 2006*, pp. 430–443. Springer (2006)
15. Rothganger, F., Lazebnik, S., Schmid, C., Ponce, J.: 3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *International Journal of Computer Vision* 66(3), 231–259 (2006)
16. Royer, E., Lhuillier, M., Dhome, M., Chateau, T.: Localization in urban environments: monocular vision compared to a differential gps sensor. In: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. vol. 2, pp. 114–121. IEEE (2005)
17. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: Orb: an efficient alternative to sift or surf. In: *Computer Vision (ICCV), 2011 IEEE International Conference on*. pp. 2564–2571. IEEE (2011)
18. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3d. *ACM transactions on graphics (TOG)* 25(3), 835–846 (2006)
19. Snavely, N., Seitz, S.M., Szeliski, R.: Modeling the world from internet photo collections. *International Journal of Computer Vision* 80(2), 189–210 (2008)
20. Sturm, P.: A historical survey of geometric computer vision. In: *Computer Analysis of Images and Patterns*. pp. 1–8. Springer (2011)
21. Waibel, M., Beetz, M., Civera, J., D’Andrea, R., Elfring, J., Galvez-Lopez, D., Haussermann, K., Janssen, R., Montiel, J.M.M., Perzylo, A., Schiessle, B., Tenorth, M., Zweigle, O., van de Molengraft, R.: Roboearth. *Robotics Automation Magazine, IEEE* 18(2), 69–82 (June 2011)
22. Yu, G., Morel, J.M.: Asift: an algorithm for fully affine invariant comparison. *Image Processing On Line* 2011 (2011)
23. Zabih, R., Woodfill, J.: A non-parametric approach to visual correspondence. In: *IEEE transactions on pattern analysis and machine intelligence*. Citeseer (1996)