Inertial sensor-aligned visual feature descriptors

Daniel Kurz, Selim Ben Himane
metaio GmbH
Infanteriestraße 19, House 4b, 80797 Munich, Germany
{daniel.kurz, selim.benhimane}@metaio.com

Abstract

We propose to align the orientation of local feature descriptors with the gravitational force measured with inertial sensors. In contrast to standard approaches that gain a reproducible feature orientation from the intensities of neighboring pixels to remain invariant against rotation, this approach results in clearly distinguishable descriptors for congruent features in different orientations. Gravity-aligned feature descriptors (GAFD) are suitable for any application relying on corresponding points in multiple images of static scenes and are particularly beneficial in the presence of differently oriented repetitive features as they are widespread in urban scenes and on man-made objects.

In this paper, we show with different examples that the process of feature description and matching gets both faster and results in better matches when aligning the descriptors with the gravity compared to traditional techniques.

1. Introduction

Many applications in the field of computer vision require finding corresponding interest points in two or more images of the same scene or object under varying viewpoints. Examples include stereo matching, camera pose estimation, and object recognition. A common way, such as described in [6], to gain such correspondences is to first extract interest points that are expected to have a high repeatability, such as corners or DoG extrema, from the individual images. The second step is then to create a local descriptor for each feature based on the intensities of its neighboring pixels. This enables its comparison and therefore its matching with its corresponding feature in an other image. The two main requirements for a good descriptor are distinctiveness, i.e. feature points corresponding to two different physical points result in different descriptors, and invariance to changes in view points and directions, illumination and image noise. This is to ensure that features corresponding to the same physical point in different images result in close descriptors with respect to a certain similarity measure.

To address the invariance to perspective distortions resulting from changes in the camera rotation, an image transformation is generally used to normalize the pixels in the region around a feature point. The descriptor is then computed based on this normalized region. It is critical that the normalized region for the same physical point in two images under different viewing angles is similar. The normalization is usually based on a feature orientation computed from the pixel intensities in the neighborhood of the feature point.

This way of orientation assignment results in problems in the presence of congruent or near-congruent features in different orientations as shown in figure 1 at the example of the corners of a window. The regions around the four corners that correspond to four different physical points would be ideally identical after normalization as shown in the left column resulting in indistinguishable descriptors.

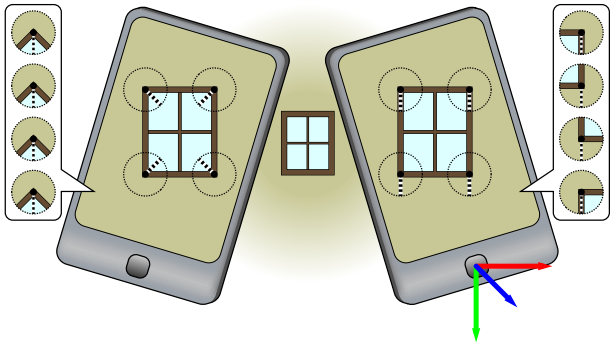

Figure 1. Schematic sketch of the effect of gravity-aligned feature descriptors (right) compared to regular techniques (left).

Increasingly more mobile devices are equipped with inertial sensors. While mobile phones already provide interfaces that make storing the gravity vector with each camera image very easy, digital cameras at least store a coarse orientation in steps of 90° with digital photos using the EXIF format which might support more precise measures in the future. As the orientation of all static objects with respect to the gravity is constant, it is worth evaluating using this orientation as a reference in the feature description process.

1.1. Contribution

This paper presents a new approach for creating local visual feature descriptors suited for static and close to upright surfaces. It outperforms standard approaches as it overcomes ambiguities resulting from congruent or near-congruent features with different orientations without constraining the camera movement. The result is an improved precision-recall characteristic. In addition, we present means to speed-up both the descriptor computation and the process of matching two or more sets of features.

The proposed technique can be applied to any existing feature descriptor based on a normalized region around the feature point and is suitable for many applications including image classification, 6-DoF camera pose tracking, and 3D scene reconstruction.

2. Related work

A variety of local feature descriptors exist, wherein a good overview and comparison is given in [7]. They all describe a feature point with a multi-dimensional vector as a function of the pixel intensities in a spatially normalized neighborhood region around the feature point. The spatial normalization is based on a feature orientation which again depends on pixel intensities in the neighborhood.

Bay et al. [2] showed that omitting the spatial normalization for Upright-SURF, outperforms regular SURF descriptors in terms of discriminative power while at the same time, the user is forced to keep the camera in an upright orientation which limits the field of possible applications.

Recently, Baatz et al. [1] described an approach to urban location recognition where they identify vanishing points in camera images and use these to rectify image parts that belong to a planar surface. They are then able to use upright feature descriptors on the rectified images that are aligned with the gravity without limiting the camera orientation. However, their approach is strongly dependent on the presence of many vertical and horizontal lines in the camera image to identify vanishing points which clearly limits the suitable environments to urban outdoor scenes.

Inertial sensors are known to deliver more stable orientation data than vision-based tracking which motivates hybrid inertial-vision tracking, e.g. [10] fuses inertial orientation data with a 6DoF pose gained from feature tracking.

In [4], a gyroscope attached to a camera is used to predict the current position of feature points that have been used in the previous frame. This enables the KLT feature tracking to cope with bigger optical flows without losing track.

To improve feature matching in catadioptric images Bazin et al. [3] measure the relative change in orientation of the camera between two images using a gyroscope and warp the second image to be aligned with the first one before matching SIFT features of the two images. They do not use the knowledge of the two images being aligned in the SIFT computation which explains why they could not report on any significant improvement when matching these highly rotational invariant feature descriptors.

Bleser and Stricker [8] present sensor fusion algorithms to predict the appearances of features by rendering a 3D model of the scene they aim to track.

The gravity measured with inertial sensors is used in [5] to automatically rectify reference images of planar areas on the ground plane or vertical surfaces. During tracking they do however not use the inertial sensor information anymore.

3. Gravity-aligned local feature descriptors

A feature descriptor is generally built such that two features corresponding to the same physical point in different images result in close descriptors with respect to a similarity measure. While two features corresponding to two different physical points are supposed to result in distinct descriptors. Often, the descriptor computation is composed of two steps namely spatial normalization of the region around the feature and the actual computation of the descriptor as a function of the normalized region [6]. Other advanced methods exist, e.g. Hessian-Affine regions [7], but in the simplest case the normalization only consists of an in-plane rotation according to the feature orientation. This orientation is defined based on pixel intensities, for instance based on the direction of the strongest gradient in a limited region around the feature. In case there are multiple dominant gradient directions the normalization and following description is carried out for all of them. The actual descriptor is finally a function of pixel intensities in the normalized region, usually based on histograms, that gives a multi-dimensional vector which will be referred to as feature descriptor.

3.1. Proposed feature orientation assignment

We align the orientation of local feature descriptors with the projection of the gravitational force in the coordinate system of the camera image. In contrast to standard approaches that gain a reproducible feature orientation from the intensities of neighboring pixels to remain invariant against camera rotation, our approach results in clearly distinguishable descriptors for congruent and near-congruent features in different orientations.

Figure 1 illustrates a window as an example of a real static object with upright surfaces. The camera of the left mobile phone captures the window and the four corners act as feature points for which a descriptor is computed. Due to invariance to rotation, as schematically illustrated with the normalized regions of the features in the left column, an ideal feature descriptor would describe these features in exactly the same way making them indistinguishable. In a real world setup, the descriptors will not be identical but

very similar and therefore virtually indistinguishable. Consequently, the probability that two features are matched and considered to correspond to the same physical point (even though they in fact correspond to two different physical points) is high for such urban scenes. Too many mismatches may result in a failure of systems using feature descriptors.

The phone on the right in figure 1 is equipped with a 3-axis-accelerometer that provides the gravity vector expressed in the device coordinate system as it is increasingly more the case for recent mobile phones. The projection of this vector in camera coordinates acts as the orientation of the four feature descriptors of the corners of the window. As illustrated in the right column the normalized regions around the four features are clearly distinct resulting in distinct feature descriptors. Thus, the proposed method reduces the probability of mismatches.

3.2. Approaches to take advantage of the gravity

The local orientation o_l of a feature computed from the intensities of neighboring pixels is usually computed such that it provides the same normalized region at any viewpoint and view direction. We propose three ways to take advantage of combining it with the global orientation o_g as the direction of the gravity, cf. figure 2 a.

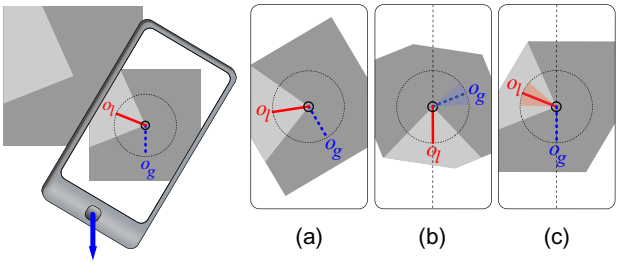

Figure 2. Each feature point has a local (red) o_l and global (blue) o_g orientation (a). Using the local orientation for descriptor alignment (b) leaves the relative global orientation as part of the descriptor. Alignment with the global orientation (c) allows to enrich the descriptor with the relative local orientation.

Regular feature descriptors with relative gravity orientation In case the global orientation corresponding to the gravity measurement o_g is too coarse, we propose to use o_l as done in regular SIFT and additionally store the relative global orientation: $o_{gl} = o_g - o_l$ for every feature as depicted in figure 2 b. Note that o_{gl} is theoretically constant and independent of the camera rotation. Similar to the sign of the Laplacian in SURF [2], we propose to use o_{gl} as a part of the descriptor and give it a higher priority. We use it to preclude the comparison of the descriptors for features whose relative orientation o_{gl} differs significantly. In fact, features with very different o_{gl} do not correspond to the same physical point. By comparing only features with a similar o_{gl}, the matching process is not only improved but also it can be speeded up significantly. In the evaluation section 4, this technique will be referred to as **regular fast**.

Gravity-aligned feature descriptors One other possibility is to use the global orientation o_g of the feature instead of the local orientation o_l as shown in figure 2 c. This allows not only to improve the matching results, but also saves computational time since the computation of o_l can be skipped. This approach will be referred to as **GAFD**.

Gravity-aligned feature descriptors with relative local orientation If the orientation of the gravity o_g can be considered accurate, we propose to use o_g as orientation for the region normalization and store o_{gl} with every feature. In case there are multiple dominant orientations o_l they are all stored relatively to the gravity o_g. Here again, in the matching process, only the descriptors of those features that have at least one similar relative orientation o_{gl} need to be compared, cf. figure 2 c. This allows a faster matching with an improved accuracy thanks to gravity-alignment. We will call this approach **GAFD fast** in section 4.

4. Evaluation and experimental results

This section reports on evaluations measuring the effect of gravity-aligned feature descriptors (GAFD) compared to regular feature descriptors both on the low-level matching performance and its impact when used in a higher level application performing image recognition on a mobile phone.

4.1. Matching precision for upright surfaces

In order to measure the impact of GAFD on the matching precision, we carried out experiments on two different mobile phones, the iPhone 3GS and the iPhone 4, providing a different level of gravity measurement accuracy. While the iPhone 3GS has a built-in 3-axis accelerometer the iPhone 4 has additionally a 3-axis gyroscope. We quantified the accuracy of the measured gravity for both devices based on 40 pictures of three vertical lines and the error distribution shows that the iPhone 4 ($\varnothing - 0.63° \pm 0.82°$) provides significantly better results than the iPhone 3GS ($\varnothing 2.8° \pm 3.67°$).

In the following experiments, we use the four planar target images that are depicted in figure 3.

Dartboard Its 20 radial sections make it a good example for many congruent features in different orientations.

Facade This urban scene has many repetitive structures both in different orientations and the same orientation.

Butterfly This is a natural (not man-made) scene.

Isetta This target represents man-made objects that do not have a large amount of repetitive features.

Figure 3. The four target images used in the matching evaluation.

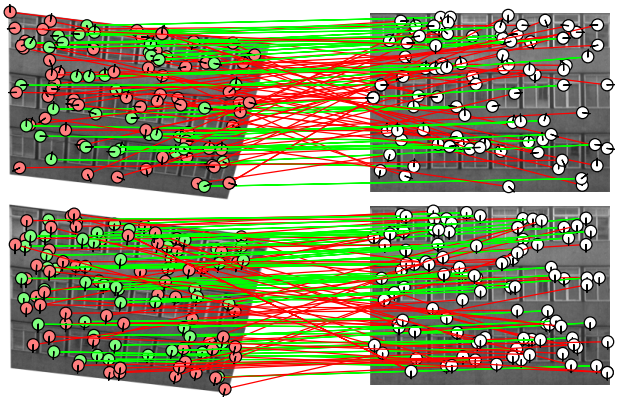

Figure 4. Correct (green) and false (red) feature matches between a query (left) and reference (right) image using SIFT regular (top) and SIFT GAFD (bottom) resulting in 15% more correct matches.

A print-out of each target image (213mm $\times$ 160mm) has been attached onto a planar surface in an upright orientation using an electronic water level. Three sequences of eight photos each have been taken of each target. In the first sequence, the mobile phone underwent strong rotations about the viewing axis (roll) and moderate rotations about the other axes while in the second sequence, the camera movement mainly consists of pitch rotations and in the third sequence yaw rotations dominate. Figure 5 shows a subset of the sequences in the top right. In addition, one picture of each target is taken from a straight perspective to act as reference image in the evaluation. As all photos have been taken with the phone in hand, some of them contain blur due to defocus or motion. With each picture, we store the measurement of the gravity vector at the time it was taken. All following computations are done offline on a PC.

First, we removed the effect of lens distortions from all images followed by resizing all query images to a resolution of (640 $\times$ 480) pixels for further processing. The four reference images have been manually rectified and resized to a resolution of (320 $\times$ 240) pixels which approximately matches the size the targets have in the query images.

The publicly available implementation of SIFT features in VLFeat [9] has been used to extract regular SIFT feature descriptors from all the images while a modified version that in addition takes the 3D gravity vector as input is used to extract gravity-aligned SIFT feature descriptors. In contrast to regular SIFT that describes a feature in multiple orientations in case there are multiple dominant gradient directions, the gravity-aligned version only describes each feature point once with the orientation of the projection of the measured gravity at the position of the feature point.

For each sequence of images the matching stage aims to find for each query feature extracted from a query image the corresponding feature in the reference image. Therefore, the nearest neighbor, i.e. the reference feature with the descriptor closest to the descriptor of the query feature is found using exhaustive search.

The ground truth to classify matches to be correct or false has been generated by manually defining the position of the four corners of the target in each query image. Based on these correspondences, a homography is computed for each query image that maps each point on the target from the query image to the reference image. For each match, this homography is used to warp the position of the query feature to the reference image. If the Euclidean distance between the warped position and the position of the matching reference feature is below a chosen threshold of 6 pixels, the match is classified as correct. Otherwise, it is considered wrong, as illustrated in figure 4 in red.

As in [7], we compare $1 - precision$ vs. $recall$ where

$$1 - precision = \frac{\#false\ matches}{\#correct\ matches + \#false\ matches}$$

$$\text{and} \quad recall = \frac{\#correct\ matches}{\#correspondences}.$$

We measure the precision-recall characteristics of four different approaches, namely **SIFT regular** and the three approaches described in 3.2: **SIFT GAFD**, **SIFT regular fast** and **SIFT GAFD fast**. In the latter two techniques, we avoid matching features whose relative orientations o_{gl} differ by more than 0.5 radians which corresponds to $\sim 29°$. Assuming equally distributed feature orientations this saves around 84% of the descriptor comparisons needed.

The samples used to plot the precision-recall curves in figure 5 were gained using different subsets of all matches whose descriptors' distance is below a threshold t for different t. Only the results for the iPhone 4 are shown since the behaviour is very similar on the iPhone 3GS although the sensor accuracy differs significantly.

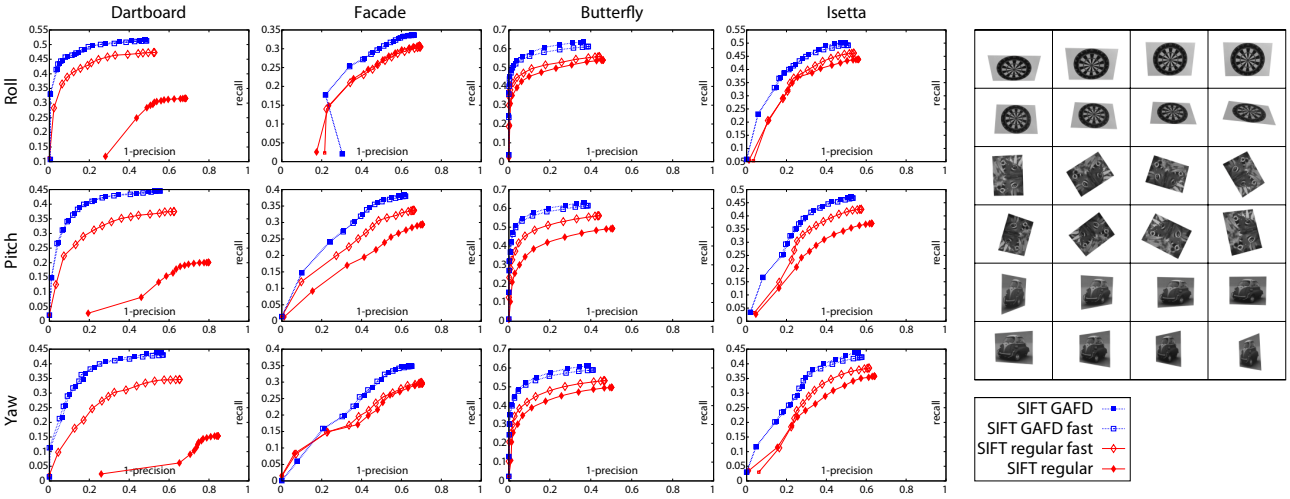

Figure 5. The precision-recall plots for the results on the iPhone 4 clearly show that all three proposed techniques outperform regular SIFT.

In nearly all cases, for all targets and for all kinds of camera movement, **SIFT GAFD** performs best closely followed by **SIFT GAFD fast**. Also, constraining the nearest neighbor search for regular SIFT features (**SIFT regular fast**) increases performance but does not give as good results as **SIFT GAFD**. As expected, the impact of gravity-alignment is particularly high for the dartboard target as it has many congruent features in different orientations. It might be surprising that the impact of GAFD is stronger in the pitch and yaw sequences than in the roll sequences. The reason is that the orientation assignment of SIFT is more invariant to roll rotations than pitch and yaw rotations as roll rotations do not change which physical points around the feature point are considered in the orientation assignment.

Considering both matching precision and computational time, **SIFT GAFD fast** gives the best results as the precision is similar to **SIFT GAFD** while it saves around 84% expensive descriptor comparisons.

4.2. Performance analysis for non-upright surfaces

While the concept described in this paper is designed to be used with vertical surfaces, this section evaluates how GAFD performs if the surfaces where features are extracted are not vertical. Therefore, the butterfly target shown in figure 3 has been attached to a tiltable surface at different orientations that were measured with an electronic water level. For each orientation, we took three images at different roll orientations of the camera keeping the camera image plane approximately parallel to the plane of the image target.

Figure 6 shows on the left the precision-recall characteristic for different angles where 90° represents a vertical surface. The right plot displays the recall for a fixed (1-precision) of 0.4 where it is visible that in this configuration **SIFT GAFD** outperforms **SIFT regular** in the range of

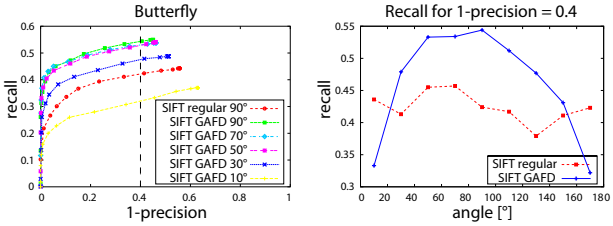

Figure 6. GAFD does not only work for vertical surfaces.

$90° \pm 60°$. Obviously, the advantage of gravity-alignment decreases more rapidly with the plane orientation differing from 90° under very steep yaw camera angles.

4.3. Object recognition on a mobile device

There are many possible applications for the GAFD such as Augmented Reality and image classification on mobile phones (see figure 7). In this paper, we measure the effect of GAFD on a higher level in a mobile museum guide. The user takes an image of an artwork, image recognition is performed on the mobile phone and information about the particular artwork is being displayed.

To this end, we randomly chose 100 artworks (paintings, drawings, photos and sculptures) in the Pinakothek der Moderne museum in Munich and took five pictures from different angles of each artwork. These pictures were stored along with the corresponding 3D gravity vector. For each artwork, we chose one picture to be the reference image while the other four images are used as query images. In contrast to the experiments shown before, here, we use software that is optimized to run on mobile phones in a reasonable amount of time. All pictures have been resized to (320×240) pixels and instead of expensive SIFT feature point extraction and descriptors, we use a fast feature de-

Figure 7. Using GAFD for object recognition (left) and Augmented Reality (right) on mobile phones.

tector and a 48-dimensional descriptor. The local orientation assignment is based on the strongest gradients similar as in SIFT. For each reference image, we extract 200 features and store them as reference features along with the ID of the artwork they belong to. From each query image, we also extract 200 features based on which we aim to detect the artwork by comparing the query features with the reference features. Since nearest neighbor search for each query feature in the set of $200 \times 100 = 20000$ reference features is too expensive on a mobile device, we use the Best-Bin-First algorithm [6] to find an approximate nearest neighbor instead. After finding it for each query feature, the detection result is eventually the ID that appears most frequently in the set of matched reference features. The entire process has been carried out both with regular and GAFD feature descriptors. Table 1 shows the percentage of correctly recognized objects in several subsets of the dataset of different sizes. We observe the detection rate increasing on average by over 15% when aligning the feature orientation with the gravity. This clearly shows an improvement over standard approaches in a real world application.

objects	recogn. rate regular	recogn. rate GAFD
40	76.88%	90.00%
60	68.33%	84.58%
80	70.94%	86.56%
100	71.25%	87.75%

Table 1. GAFD improves the recognition rates of a mobile guide.

5. Conclusions and outlook

The evaluations in the last section, clearly shows the improvement of GAFD both on the precision of matches and in an application on a mobile phone. The proposed method outperforms classical approaches without introducing any drawbacks since even the computational costs are reduced. Of course, the field of applications for GAFD is limited to the presence of an inertial sensor and does not work for features on surfaces that are parallel to the ground plane or close to it. However, we believe that there is a variety of applications where GAFD are useful. Increasingly more devices are equipped with inertial sensors and many tasks, particularly outdoor self-localization, rely mainly on features that are located on facades, i.e. vertical surfaces.

Our **regular fast** approach can also be applied to information retrieval on image databases, such as flickr, storing a coarse orientation with respect to gravity in their EXIF data.

The concept of aligning feature descriptors with a common coordinate system is not limited to the gravity as described in this paper but has a variety of possible implementations. Many mobile phones include a compass that indicates the direction of the north. Combining this with the gravity results in the full 3DoF orientation in absolute world coordinates. This again allows to align features with any absolute orientation in world space and overcomes the constraint that surfaces must not be parallel to the ground plane. In outdoor self-localization tasks, we envision to align features with multiple reference orientations and choose the appropriate alignment online based on the device orientation. Besides sensors that provide an absolute orientation any tracking system attached to a camera providing relative transformations between the individual camera images can be used to align feature descriptors with a common orientation in applications that do not require any offline learned model of the environment, such as 3D reconstruction.

References

[1] G. Baatz, K. Koser, D. Chen, R. Grzeszczuk, and M. Pollefeys. Handling urban location recognition as a 2d homothetic problem. In *Proc. ECCV*, 2010. 162

[2] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. *CVIU*, 110(3):346–359, 2008. 162, 163

[3] J. C. Bazin, I. Kweon, C. Demonceaux, and P. Vasseur. Improvement of feature matching in catadioptric images using gyroscope data. In *Proc. ICPR*, 2008. 162

[4] M. Hwangbo, J.-S. Kim, and T. Kanade. Inertial-aided KLT feature tracking for a moving camera. In *Proc. IROS*, 2009. 162

[5] W. Lee, Y. Park, V. Lepetit, and W. Woo. Point-and-Shoot for Ubiquitous Tagging on Mobile Phones. In *Proc. ISMAR*, 2010. 162

[6] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 161, 162, 166

[7] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *Trans. PAMI*, 27(10):1615–1630, 2005. 162, 164

[8] G. B. D. Stricker. Advanced tracking through efficient image processing and visual-inertial sensor fusion. *C&G*, 33:59–72, 2009. 162

[9] A. Vedaldi and B. Fulkerson. Vlfeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org, October 2010. 164

[10] S. You, U. Neumann, and R. Azuma. Hybrid inertial and vision tracking for augmented reality registration. In *Proc. VR*, 1999. 162