

Associating locations from wearable cameras

Jose Rivera-Rubio

<http://www.bicv.org>

Ioannis Alexiou

i.alexiou09@imperial.ac.uk

Anil Bharath

a.bharath@imperial.ac.uk

Riccardo Secoli

r.secoli@imperial.ac.uk

Luke Dickens

luke.dickens@imperial.ac.uk

Emil C. Lupu

e.c.lupu@imperial.ac.uk

Imperial College London

South Kensington Campus, UK

Abstract

In this paper, we address a specific use-case of wearable or hand-held camera technology: indoor navigation. We explore the possibility of crowdsourcing navigational data in the form of video sequences that are captured from wearable or hand-held cameras. Without using geometric inference techniques (such as SLAM), we test video data for navigational content, and algorithms for extracting that content. We do not include tracking in this evaluation; our purpose is to explore the hypothesis that visual content, on its own, contains cues that can be mined to infer a person's location. We test this hypothesis through estimating positional error distributions inferred during one journey with respect to other journeys along the same approximate path.

The contributions of this work are threefold. First, we propose alternative methods for video feature extraction that identify candidate matches between query sequences and a database of sequences from journeys made at different times. Secondly, we suggest an evaluation methodology that estimates the error distributions in inferred position with respect to a ground truth. We assess and compare standard approaches from the field of image retrieval, such as SIFT and HOG3D, to establish associations between frames. The final contribution is a publicly available database comprising over 90,000 frames of video-sequences with positional ground-truth. The data was acquired along more than 3 km worth of indoor journeys with a hand-held device (Nexus 4) and a wearable device (Google Glass).

1 Introduction

Self-localization within an indoor space has numerous real-world applications, ranging from navigation inside public spaces and large shopping and social environments to assistive de-

vices for people with visual impairment. Harvesting information from radio-strength signals and radio beacons to perform localization is an emerging technology. However, few potential solutions are as compelling as those using visual information, captured from wearable or hand-held cameras, and conveyed into knowledge about how to navigate a space.

This work proposes an alternative approach to geometric and SLAM-based localization. Location is, instead, associated through visual queries against the paths of other users, rather than by explicit map-building or geometric inference. We test this idea in a new dataset of *visual paths* [17], containing more than 3 km of video sequences captured through *multiple* passes along 10 corridors in a large building with ground truth. We compare custom-designed descriptors with SIFT [2] and HOG3D [9]. Standard Bag-of-Visual Words (BoVWs) approaches are used to index and associate views between journeys. The results suggest that, even without tracking, significant cues about localization can be captured and used to infer location. The application to wearable camera technology – whereby image cues are harvested from volunteered journeys, then used to help other users of the same space navigate – is the eventual goal of this work, which is a natural extension to recently reported approaches based on harvesting environmental signals [25].

2 Related work

Matching between visual paths We define a *visual path* as a collection of image frames that are induced by the relative motion of a person in a scene. The work reported in this paper involves matching the visual paths of a “new” journey instance to previous, similar instances.

Early work by Matsumoto et al. [13] introduced a similar concept of the “view-sequenced route representation”. In this scheme, a robot could perform simple navigation tasks by correlating current views against those held in a database. Ohno et al. [14] also worked on this idea, using the difference between frames of detected vertical lines to estimate changes in position and orientation. Their results were constrained to controlled robot movement, and therefore arguably of limited applicability to images obtained from human ego-motion. Also employing vertical lines as features, this time from omni-directional images, Tang *et al.* used estimated position differences between sequences to perform robot navigation [21]. To make the inference more robust, they used recorded odometry at training time. This approach would certainly reduce the error in the localization task. However, it could lead to “solving” the training route, without truly analysing the performance of feature matching methods. Furthermore, without ground-truth available in a crowdsensing setting, the technique of training with ground-truth is of limited usability. On the other hand, with many passes through the same space, the reference for a journey could be the visual paths themselves. In this case, we would use ground-truth – if available – only to ascertain the accuracy of proposed matching or localization methods. This is the approach taken in our current work.

The performance of previously reported methods that use a retrieval-type approach, albeit of the order of tens of cm, cannot be taken as representative for the evaluation of the methods presented in this paper. The reviewed publications report results in routes of a few metres in length. Our evaluation is in a dataset three orders of magnitude longer. Deliberately, we do not include any tracking, such as Kalman filtering, which can often hide poor measurement performance.

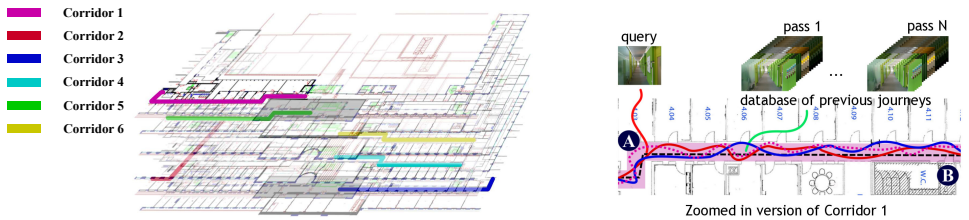


Figure 1: Maps of the recording locations (left). A sample path (Corridor 1, C1) with the multiple passes overlaid (right). Each of these passes represents a database sequence.

Crowdsourcing visual paths. Our usage setting represents a particularly data-intensive form of crowdsensing in which the image streams from wearable cameras could be volunteered to others as reference paths for indoor journeys. An illustration of this concept is presented in Fig. 1.

This type of crowdsensing approach is gaining interest, with remarkable work from Google’s indoor localization systems and crowdsourced sensor information and maps [8]. In terms of a retrieval-based visual localization system, the NAVVIS team [9] released a dataset for evaluating indoor navigation from a camera-equipped robot. They also advanced earlier work on visual localization based on matching of SIFT descriptors [15] to one using a bag of features that could be stored in mobile phones for quick retrieval [18, 19]. The dataset we introduce in this work is not constrained to robot navigation, as it includes the ego-motion associated with hand-held and wearable devices.

Alternative methods: non feature-based and sensor merging. For outdoor navigation, the Global Positioning System (GPS) has been in widespread use for many years. In an *indoor* context, localization technology is still rapidly evolving [16, 20, 25]. Using visual information is towards the higher end of computational complexity, and possibly the lower-end of reliability; one would certainly seek to support this approach with other forms of sensor such as Received Signal Strength Indication (RSSI) data, magnetometers, and tracking algorithms [16, 18, 19]. In this paper, we seek to explore efficient techniques that could be used to index and compare the visual path information gathered by multiple user journeys, and to measure the potential of vision on its own as a localization mechanism.

A biological intuition. Another source of our motivation for the idea of retrieval-based localization is supported by the well-characterized biological hippocampal place cells [9] that recognise a location from sensory inputs that include those captured by an animal’s eyes. This does *not* suggest that techniques based on optical flow are not relevant: rather, the striking conclusion from recent research is that multiple approaches to visual location inference are at work in biological systems, including optic flow [10], and other mechanisms that may not explicitly involve brain areas specialized in visual motion computation [9].

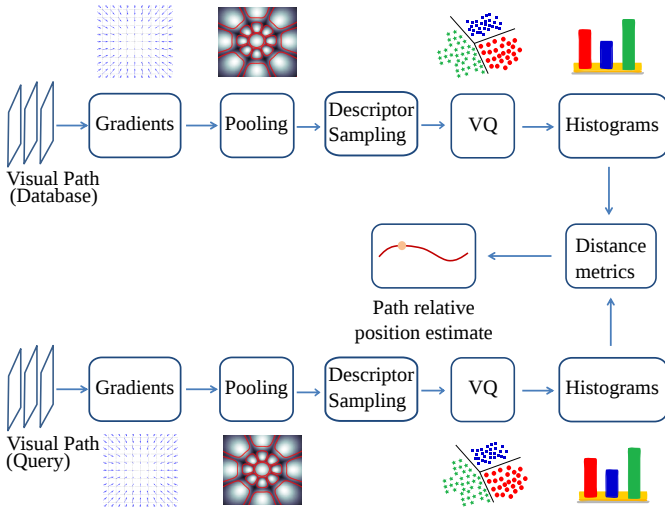


Figure 2: The stages in processing image sequences from database and query visual paths are illustrated above. This does not show the process behind the estimation of ground-truth for the experiments, which is described separately in Section 4. Variants of the gradient and pooling operators, quantization approaches and distance metrics are described in Section 3.

3 Methods

3.1 Pipeline

We evaluated the performance of several approaches to matching image queries taken from one visual path against the remainder of the visual paths. In order to index and query the visual path datasets, we adopted a sequence of processes that is illustrated in Fig. 2. We describe the details behind each of the processes (e.g. gradient estimation, spatial pooling) in Section 3.2. We considered descriptors that operate on *single* frames (spatial) as well as descriptors that operate on *multiple* frames (spatio-temporal).

3.2 Local descriptors

Keypoint based SIFT (KP_SIFT). The original implementation of Lowe’s SIFT descriptor follows the extraction of interesting points in the image that are stable to certain transformations, the “SIFT keypoints” [12]. This is widely used across many branches of computer vision, from object recognition to motion detection and SLAM. We used the standard implementation from VLFEAT [23] to compute $\vec{\nabla}f(x, y; \sigma)$ where $f(x, y; \sigma)$ represents the embedding of image $f(x, y)$ within a Gaussian scale-space at scale σ . We set the parameter *PeakThresh* to 0 to filter out small local maxima in scale-space.

Dense SIFT (DSIFT). The Dense-SIFT (DSIFT) descriptor [13] is a popular alternative to keypoint based SIFT. It sacrifices some invariance properties available with keypoint-based SIFT, producing descriptors that are densely, rather than sparsely, distributed across the image. This DSIFT descriptor was calculated by sampling of the smoothed estimate of

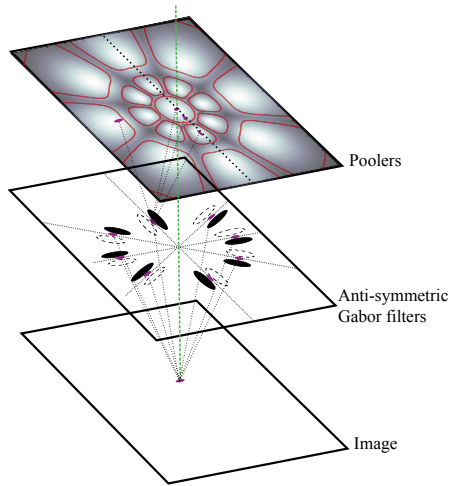


Figure 3: The spatial pooling pattern used for single frame Gabor filtering is based on the regions shown here. These regions were generated by sampling Eq. (1) to create pooling masks. The masks can be applied to the Gabor filtered video frame outputs by spatial convolution, followed by sub-sampling the output every 3 pixels. See text for further details.

$\vec{\nabla}f(x, y; \sigma)$. We used the implementation of the VLFEAT toolbox, setting $\sigma = 1.2$, with a stride length of 3 pixels. This yielded around 2,000 descriptors per frame, each describing a patch of roughly 10×10 pixels.

Single Frame Gabor descriptors (SF_GABOR). An alternative single frame technique based on a tuned, odd-symmetric Gabor-based descriptor is the SF_GABOR. For this, we used 8-directional spatial Gabor filters previously tuned on PASCAL VOC data [5] in order to provide an implicit encoding of the orientation of local image structures. Each filter gives rise to a filtered image plane, denoted $\mathbf{G}_{k,\sigma}$. For each plane, we compute the discrete spatial convolution, $\mathbf{G}_{k,\sigma} * \Phi_{m,n}$, with a series of pooling functions, $\Phi_{m,n}$. The latter are produced by spatial sampling of the function:

$$\Phi(x, y; m, n) = e^{-\alpha \left[\log_e \left(\frac{x^2 + y^2}{d_n^2} \right) \right]^2 - \beta |\theta - \theta_m|} \quad (1)$$

with $\alpha = 4$ and $\beta = 0.4$. The values of m and n were chosen to produce 8 angular regions at each of two distances d_1, d_2 away from the centre of a spatial pooling region. For the central region, corresponding to $m = 0$, there was no angular variation but instead a log-normal radial decay, with a limiting value at $(x, y) = (0, 0)$. This arrangement yielded a total of 17 spatial pooling regions. The resulting 17×8 fields are sub-sampled to produce dense 136-dimensional descriptors, each representing an approximate 10×10 region, and yielding around 2,000 descriptors per image frame after spatial sub-sampling.

Space-Time descriptors. Given the potential richness available from space-time information, we explored three distinct approaches to generate space-time patch descriptors. When

generating the descriptor associated with each patch, all approaches yield multiple descriptors per frame, and all take into account neighbouring frames in time. In contrast to a sparse-sampling approach of a keypoint-based descriptor, all three densely sample the video sequence. The three methods are i) HOG 3D [9]; ii) a space-time, antisymmetric Gabor filtering process (ST_GABOR); and iii) a Spatial Derivative, Temporal Gaussian (ST_GAUSS) filter.

1. The **HOG 3D** descriptor (HOG3D) [9] was introduced with the aim of extending the very successful two-dimensional histogram of oriented gradients technique [4], to space-time fields, in the form of video sequences. HOG 3D seeks computational efficiencies by smoothing using box filters, rather than Gaussian spatial or space-time kernels. This allows three-dimensional gradient estimation across multiple scales using *integral video* representations, a direct extension of the integral image idea [24]. The gradients from this operation are usually performed across multiple scales. We used the dense HOG 3D option from the implementation of the authors, and the settings yielded approximately 2,000 descriptors per frame of video. Each descriptor contained 192 elements.
2. **Space-time Gabor (ST_GABOR)** functions have been used in activity recognition, structure from motion and other applications [11]. We performed one dimensional convolution between the video sequence and three one-dimensional Gabor functions along either one spatial dimension i.e. x or y , or along t . The one-dimensional convolution is crude, but appropriate if the videos have been downsampled. The spatial extent of the Gabor function was set to provide one complete cycle of oscillation over approximately 5 pixels of spatial span, both for the x and y spatial dimensions. The filter for the temporal dimension was set to provide around one oscillation over 9 frames. We also explored symmetric Gabor functions, but found them rather less favourable.

After performing three separate filtering operations, each pixel of each frame is assigned a triplet of values corresponding to the result of each filtering operation. The three values are treated as being components of a 3D vector. Over a spatial extent of around 16×16 pixels taken at the central frame of the 9-frame support region, these vectors contribute weighted votes into descriptor bins according to their azimuth and elevations, with the weighting being given by the length of the vector. The votes are also partitioned according to the approximate spatial lobe pattern illustrated in Fig. 3. Each frame had approximately 2,000 ST_GABOR descriptors, each of 221 elements.

3. A final variant of space-time patch descriptor was designed. This consisted of spatial derivatives in space, combined with smoothing over time (**ST_GAUSS**). In contrast to the strictly one-dimensional filtering operation used for the ST_GABOR descriptor, we used two 5×5 gradient masks for the x and y directions based on derivatives of Gaussian functions, and an 11-point Gaussian smoothing filter in the temporal direction, using a standard deviation of 2. 8-directional quantization was applied to the angles of the gradient field, and a voting process incorporating gradient magnitude was used to distribute votes across the bins of a 136-dimensional descriptor. Like the ST_GABOR descriptor, pooling functions, similar to those shown in Fig. 3, were applied. The number of descriptors produced was the same as for the other methods using patch-level descriptions.

3.3 Quantization and histogram encoding

For the approaches described in Section 3.2, hard assignment (HA) was applied to assign descriptors to a dictionary term. The dataset was partitioned by selecting $M - 1$ of the M video sequences of passes through each possible path. These $M - 1$ sequences have a total of N frames. A dictionary of visual words was created by running the k -means algorithm on the partitioned set of training descriptors contained in the N frames. We fixed the dictionary size to 4,000 in order to achieve a balance between computational time and atom stability, and allowing comparison with the work of others [9].

The resulting dictionaries were then used to encode the descriptors of the $M - 1$ training passes and the remaining query pass. First, the descriptors found in every frame were each assigned to the nearest visual word using a Euclidean distance metric. Secondly, the frequency of occurrence of the dictionary words (or atoms) for every frame was used to create a histogram representing each frames in the training database, and the same process was used to encode each possible query frame from the remaining path (which was *not* used to build the dictionary). These histograms were all L_2 -normalised.

3.4 Localization using histogram distances

Once histograms had been produced, a distance measurement was used to compare the similarity of histograms in a query frame with the database entries. The query operation was simply performed by using the kernel approaches described in [23]. We used the χ^2 kernel; other kernels such as the Hellinger, are possible, but the χ^2 option appeared to work best in the tests we conducted. For the $M - 1$ videos captured over each path in the database, the queries were constructed from the remaining path. Each query frame, H_q , resulted in $M - 1$ separate comparison vectors containing scores. By using these kernel-based comparisons (which are always positive, and act as the inverse of a distance metric), we identified the best matching frame, \hat{f} , from pass, \hat{p} , across all of the $M - 1$ vectors. This may be expressed as:

$$L(\hat{p}, \hat{f}) = \operatorname{argmax}_{p,f} \{K_D(H_q, H_{p,f})\} \quad (2)$$

where $H_{p,f}$ denotes the series of normalised histogram encodings, indexed by p drawn from the $M - 1$ database passes, and f denotes the frame number within that pass. K_D denotes the so-called “kernelized” version of distance measure [23]. To measure the localization error, we used the ground-truth estimates that were acquired at the same time as the videos. The estimated position of a query, L , was simply taken to be that of the best match given by Eq. (2). However, in a more robust implementation, checks could be done that would require similar matches in neighbouring frames, both in query and pass.

4 Experiments

4.1 Data acquisition and ground truth

A total of 60 videos were acquired from 6 corridors of a large building. Two different devices were used for the acquisition, with 30 videos each. One was an LG Google Nexus 4 phone running Android 4.4.2. The video data was acquired at approximately 24-30 fps at two different resolutions, 1280×720 and 1920×1080 pixels. Google Glass (Explorer edition) was used at a resolution of 1280×720 , at a frame rate of 30 fps. A surveyor’s wheel (Silverline)







	Photo	Length (m)			No. of frames		
		Avg	Min	Max	Avg	Min	Max
C1		57.9	57.7	58.7	2157	1860	2338
C2		31.0	30.6	31.5	909	687	1168
C3		52.7	51.4	53.3	1427	1070	1777
C4		49.3	46.4	56.2	1583	1090	2154
C5		54.3	49.3	58.4	1782	1326	1900
C6		55.9	55.4	56.4	1471	1180	1817
Total		3.042 km			90,302 frames		

Table 1: A summary of the dataset with thumbnails.

with a precision of 10 cm and error of $\pm 5\%$ was used to record distance, but was modified by wiring its encoder to a Raspberry Pi running a number of measurement processes. The Pi was synchronised to network time, enabling synchronisation with timestamps in the video sequence. Because of the variable-frame rate of acquisition, timestamp data from the video was used to align ground-truth measurements with frames. This data was used to assess the accuracy of associating positions along journeys through frame indexing and comparison.

The dataset contains 3.05 km of journey data acquired at a casual indoor walking speed. For each corridor, ten passes (i.e. 10 separate visual paths) were obtained. Five of these videos were acquired with the hand-held Nexus, and the remainder with Glass. Table 1 summarises the acquisition. The length of the sequences varies, due to a combination of different walking speeds and/or different frame rates and corridor lengths. A combination of daylight/nighttime acquisitions was also performed, and prominent windows occasionally introduced strong lighting in some portions of the videos. Variations are observable in some of the corridors from one pass to another, due to physical changes and occasional appearances from people walking along. In total, more than 90,000 frames of video were labelled with positional ground-truth in a path-relative fashion. The dataset is publicly available at [17].

4.2 Error distributions

We estimated localization error distributions in order to quantify the accuracy of being able to associate *locations* along physical paths in corridors within the dataset described in Section 4.1. By permuting the paths that are used as reference journeys, and by randomly selecting query images from the remaining path, we are able to estimate the error in localization. Repeated runs with random selections of groups of frames allowed the variability in these estimates to be obtained. This includes effects that might be due to different paths being selected as the reference set. To estimate the error distributions, we measured the absolute error in localization as a distance, ε , relative to the ground truth for that route. These errors are provided as estimates of $P(\varepsilon < x)$. We used the ground-truth information acquired as described in Section 4.1.

In Figs. 4a to 4f, we provide separate assessments of the *variability* in error distribution when 1 million permuted queries are performed; these were obtained by cycling through 1,000 permutations of 1,000 randomly selected queries. In Fig. 4g, we compare the error distributions of all techniques. For long distances, the CDFs of error for all methods ap-

proaches unity; we thus only show a close-up of the interval $[0, 25]$ m.

All the results were generated with a downsampled version of the videos at 208×117 pixels; these are also supplied with the dataset.

5 Results

We calculated the average absolute positional error (in metres) and the standard deviation of the absolute positional errors across the provided dataset, and these are shown in Table 2. For these errors, all queries, by a leave-one-out strategy, have been used, but there is otherwise no random sampling of the queries. Standard deviations of the absolute errors are also provided. Table 2 also provides the Area-Under-Curve (AUC) values obtained from the CDFs of Fig. 4g.

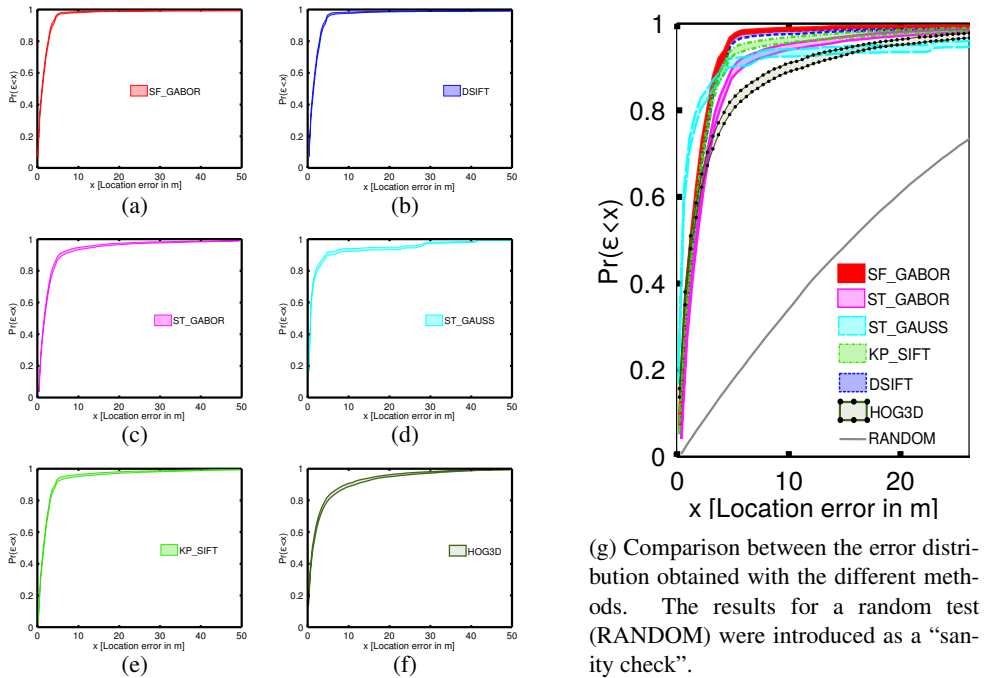


Figure 4: Cumulative Distribution Functions of the methods under study.

The results show that localization is achieved with good accuracy in terms of CDF and AUC without a large difference between the applied methods, despite the big diversity in their complexity. Absolute errors show significant differences between methods, with average absolute errors in the range of 1.5 m to 4.20 m. Single frame methods (SF_GABOR, KP_SIFT and DSIFT) perform slightly better than spatio-temporal approaches. This is not surprising, as the spatio-temporal methods might be strongly affected by the self motion over fine temporal scales.

In spite of using image retrieval methods in isolation, the attainable accuracy appears to be in line with those of other methods reviewed in Section 2. However, previously reported

Method	Error summary (m)		AUC (%)	
	μ_ϵ	σ_ϵ	Min	Max
SF_GABOR	1.59	0.11	96.11	96.39
DSIFT	1.62	0.11	95.96	96.31
KP_SIFT	2.14	0.17	94.58	95.19
ST_GAUSS	2.11	0.24	94.82	95.57
ST_GABOR	2.54	0.19	93.90	94.44
HOG3D	4.20	1.33	90.89	91.83

Table 2: Summaries of average absolute positional error and standard deviation of positional errors for different descriptor types. μ_ϵ is the average absolute error, and σ_ϵ is the standard deviation of the error in metres. Top: single frame methods. Bottom: spatio-temporal methods.

methods include tracking, the use of other sensors, or estimates of motion. In this work, no form of tracking was used in estimating position: this was deliberate, in order that we could assess performance in inferring location from the visual data fairly. Introducing tracking will, of course, improve localization performance, and could reduce query complexity. Yet, tracking often relies on some form of motion model, and for pedestrians carrying or wearing cameras, motion can sometimes be relatively unpredictable.

6 Conclusion

We have presented three main contributions to the topic of indoor localization using visual path matching from wearable and hand-held cameras. We provide an evaluation of six local descriptor methods: three custom designed and three standard image (KP_SIFT and DSIFT) and video (HOG3D) matching methods as baseline. These local descriptions follow a standard bag-of-words and kernel encoding pipeline. The code for both the local descriptors and for the evaluation pipeline is available on the web page [17]. We also make available a large dataset with ground truth of indoor journeys to complete the evaluation framework.

The results show that there is significant localization information in the visual data, and that errors as small as 1.6 m over a 50 m distance can be achieved, even without tracking. We have reported the results in two ways: a) average absolute positional errors, and b) error distributions, both of which allow image descriptions to be assessed for their localization capability. The latter could also be used to build a measurement model for inclusion in a Kalman or particle filter aimed at supporting human ambulatory navigation.

We plan to introduce tracking in future work. There are, of course, numerous other enhancements that one could make for a system that uses visual data; integration of data from other sensors springs to mind, such as inertial sensing, magnetometers and RSSI. Although fusing independent and informative data sources would theoretically lead to improvements in performance, we would argue that the methods applied to infer location from each information source should be rigorously tested, both in isolation and as part of an integrated system. This would help ensure that real-world systems would be somewhat robust to sensor failure. We anticipate that using vision to associate locations in the journeys of several users through their visual paths could play an important role in navigation.

References

- [1] M. Bregonzio. Recognising action as clouds of space-time interest points. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1948–1955, June 2009. doi: 10.1109/CVPR.2009.5206779. URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5206779>.
- [2] Neil Burgess, Eleanor A Maguire, and John O’Keefe. The human hippocampus and spatial and episodic memory. *Neuron*, 35(4):625–641, 2002.
- [3] Ken Chatfield, Victor Lempitsky, Andrea Vedaldi, and Andrew Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *Proceedings of the British Machine Vision Conference (BMVC)*, number 1, pages 76.1–76.12. British Machine Vision Association, 2011. ISBN 1-901725-43-X. doi: 10.5244/C.25.76. URL <http://www.bmva.org/bmvc/2011/proceedings/paper76/index.html>.
- [4] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, 1:886–893, 2005. doi: 10.1109/CVPR.2005.177. URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1467360>.
- [5] Mark Everingham, Luc Gool, Christopher K I Williams, John Winn, and Andrew Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2):303–338, 2009. ISSN 09205691. doi: 10.1007/s11263-009-0275-4. URL <http://www.springerlink.com/index/10.1007/s11263-009-0275-4>.
- [6] Tom Hartley, Colin Lever, Neil Burgess, and John O’Keefe. Space in the brain: how the hippocampal formation supports spatial cognition. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369(1635):20120510, 2014.
- [7] R Huitl and G Schroth. TUMindoor: An extensive image and point cloud dataset for visual indoor localization and mapping. In *International Conference on Image Processing*, 2012. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6467224.
- [8] Waleed Kadous and Sarah Peterson. Indoor Maps : The Next Frontier. In *Google IO*, 2013.
- [9] A Kläser, M Marszalek, and Cordelia Schmid. A spatio-temporal descriptor based on 3D-gradients. In *British Machine Vision Conference*, pages 995—1004, 2008. URL <http://eprints.pascal-network.org/archive/00005039/>.
- [10] Oliver W. Layton and N. Andrew Browning. A Unified Model of Heading and Path Perception in Primate MSTd. *PLoS Computational Biology*, 10(2):e1003476, February 2014. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1003476. URL <http://dx.plos.org/10.1371/journal.pcbi.1003476>.

- [11] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178. Ieee, 2006. ISBN 0-7695-2597-0. doi: 10.1109/CVPR.2006.68.
- [12] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004. URL <http://www.springerlink.com/index/H4L02691327PX768.pdf>.
- [13] Yoshio Matsumoto, M Inaba, and H Inoue. Visual navigation using view-sequenced route representation. In *International Conference on Robotics and Automation*, number April, pages 83–88, 1996. ISBN 0-7803-2988-0. doi: 10.1109/ROBOT.1996.503577. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=503577.
- [14] T. Ohno, A. Ohya, and S. Yuta. Autonomous Navigation for Mobile Robots Referring Pre-recorded Image Sequence. In *IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS '96*, volume 2, pages 672–679. Ieee, 1996. ISBN 0-7803-3213-X. doi: 10.1109/IROS.1996.571034. URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=571034>.
- [15] SY Park, SC Jung, YS Song, and HJ Kim. Mobile robot localization in indoor environment using scale-invariant visual landmarks. In *18th IAPR International Conference in Pattern Recognition*, pages 159–163, 2008. URL <http://www.eurasip.org/Proceedings/Ext/CIP2008/papers/1569094833.pdf>.
- [16] M Quigley and D Stavens. Sub-meter indoor localization in unmodified environments with inexpensive sensors. In *Intelligent Robots and Systems*, pages 2039–2046. Ieee, October 2010. ISBN 978-1-4244-6674-0. doi: 10.1109/IROS.2010.5651783.
- [17] Jose Rivera-Rubio, Ioannis Alexiou, and Anil A. Bharath. RSM dataset, 2014. URL <http://rsm.bicv.org>.
- [18] Georg Schroth and Robert Huitl. Mobile visual location recognition. *IEEE Signal Processing Magazine*, (July):77–89, 2011. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5888650.
- [19] Georg Schroth and Robert Huitl. Exploiting prior knowledge in mobile visual location recognition. In *IEEE ICASSP*, pages 4–7, 2012. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6288388.
- [20] Guobin Shen, Zhuo Chen, P Zhang, Thomas Moscibroda, and Yongguang Zhang. Walkie-Markie: Indoor Pathway Mapping Made Easy. In *10th USENIX Symposium on Networked Systems Design and Implementation (NSDI '13) USENIX*, pages 85–98, 2013. URL http://research.microsoft.com/en-us/um/people/moscitho/Publications/NSDI_2013.pdf.
- [21] Lixin Tang and S Yuta. Vision based navigation for mobile robots in indoor environment by teaching and playing-back scheme. In *International Conference on Robotics and Automation*, pages 3072–3077, 2001. ISBN 0780364759. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=933089.

- [22] A Vedaldi and B Fulkerson. VLFeat: An Open and Portable Library of Computer Vision Algorithms, 2008. URL <http://www.vlfeat.org/>.
- [23] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010.
- [24] Paul Viola and Michael Jones. Robust real-time object detection. *International Journal of Computer Vision*, pages 1–25, 2001. URL <http://www.staroceans.net/documents/CRL-2001-1.pdf>.
- [25] He Wang, S Sen, Ahmed Elgohary, M Farid, and M Youssef. Unsupervised Indoor Localization. In *MobiSys*. ACM, 2012. ISBN 9781450313018. URL <http://synrg.ee.duke.edu/papers/unloc.pdf>.