Noname manuscript No.
(will be inserted by the editor)

Flyphone: Visual Self-Localisation Using a Mobile Phone as Onboard Image Processor on a Quadrocopter

Sara Erhard · Karl E. Wenzel · Andreas Zell

Received: date / Accepted: date

Abstract An unmanned aerial vehicle (UAV) needs to orient itself in its operating environment to fly autonomously. Localisation methods based on visual data are independent of erroneous GPS measurements or imprecise inertial sensors.
In our approach, a quadrocopter first etablishes an image database of the environment. Afterwards, the quadrocopter is able to locate itself by comparing a current image taken of the environment with earlier images in the database. Therefore, characteristic image features are extracted which can be compared efficiently. We analyse three feature extraction methods and five feature similarity measures. The evaluation is based on two datasets recorded under real conditions.
The computations are performed on a Nokia N95 mobile phone, which is mounted on the quadrocopter. This lightweight, yet powerful device offers an integrated camera and serves as central processing unit. The mobile phone proved to be a good choice for visual localisation on a quadrocopter.

Keywords Computer vision · Unmanned aerial vehicles (UAV) · Visual localisation · Mobile devices · Onboard computation

1 INTRODUCTION

This paper introduces a visual self-localisation system for unmanned aerial vehicles (UAV) which is efficient enough to run at 1.7 Hz on a Nokia N95 mobile phone. Our UAV is planned to operate autonomously from external control: It is designed to navigate without a pilot and is to be independent of a base station's processing power or memory. Further, we focus on vision-based localisation without GPS or inertial sensors. The environment is not artificially changed by markers.
We use a model quadrocopter as UAV. The system is limited to a payload of only 300 g. Thus, we were looking for a lightweight device with a camera and enough processing power and memory. We decided to use a mobile phone. It weighs only 120 g and meets the above criteria.

S. Erhard · K. E. Wenzel · A. Zell
Department of Computer Science, University of Tübingen, Sand 1, 72076 Tübingen, Germany
E-mail: {sara.erhard, karl.e.wenzel, andreas.zell}@uni-tuebingen.de

Our approach is based on *content-based image retrieval* (CBIR). That is, we build an image database from the environment during one exploration flight. Afterwards, the quadrocopter is able to locate itself by comparing the current image to the training database. Therefore, the actual localisation is only based on visual data. However, at this stage, we use GPS as ground truth at the exploration flight.
To compare images, we extract features which identify and characterize an image. The main concern of appearance-based localisation methods is to find unique image attributes that represent the view efficiently and in a simple data structure [27]. Finding features containing these characteristics is a difficult task due to the complex nature of the environment of an outdoor robot. Shapes are unstructed and irregular, colors are permanently changing. Illumination alters depending on time of day and weather conditions. People or objects like cars can change their position. In our analysis we lay our focus on the influence of illumination changes. Further, the camera is mounted on a quadrocopter, which implies that its pose can vary in six degrees of freedom (DoF). Therefore, the features should not be influenced by image rotations and translations.
One distinguishes between global methods, which use the whole image to calculate features, and local methods, which use several points of interest. The detection of salient points is computationally expensive. Multiple local features have to be stored for each image. A global feature establishes a compact representation of one image. Besides simplicity, it requires less memory in comparison with local features. Its main disadvantage is its sensitivity to occlusion. In order to archieve robustness while still being efficient, we use local image grids to handle the occlusion problem. We analyse three global methods under real conditions. To compare these image features, one needs distance measures to determine how similar two features are. Thus, we analyse five different comparison measures.
We combine the two aspects of visually locating and mobile computing to an UAV system which we call "Flyphone". The focus of this paper lies mainly on analysing different feature extraction algorithms for localisation purposes.

The remainder of the paper is organized as follows. In Section 2, we introduce the related work. Section 3 describes our robot system, consisting of a quadrocopter and a mobile phone. We present algorithms, which extract image features, and similarity measures to compare these features in Section 4. Section 5 analyses the presented extraction and comparison methods with our airborne system. Finally, conclusions are drawn in Section 6.

2 RELATED WORK

One contribution of this work is to bring together UAVs and mobile phones as lightweight, but computationally powerful processing devices. Thus, related work can be categorised into two main areas: Self-localisation of robots, especially UAVs, and image processing on hand-held mobile phones.
Autonomous aerial vehicles have become an active area of research in recent years. Some approaches give a solution to the problem of self-localisation by realizing *Simultaneous Localisation And Mapping* (SLAM) with visual data and additional sensors [11, 18, 19, 24, 25]. It aims at simultaneously building and updating an environment map. The problem of determining the position and orientation of a robot by analysing the associated camera images is known as *visual odometry* in the robot vision commu-

nity. Accurate results have been obtained by using stereo or omnidirectional cameras [6, 14, 20]. Methods like optical flow analysis or local feature tracking are used for instance on Mars missions [14]. Not only for localisation, but also for stabilisation or control purposes various approaches use artificial markers or known targets [2, 16, 17]. By contrast, we analyse images by extracting features of the environment without artificial markers or targets. This is also done in work with groundrobots [1, 9, 10, 21, 23, 29, 33]. Instead of an artificially modified environment, we need some prior knowledge with means of raw image data. This knowledge is obtained by preceding exploration flights, where images of the environment are recorded. Splitting the process in two phases has also been done by Valgren *et al.* [29], Lamen *et al.* [12], Ulrich *et al.* [28] and Zhou *et al.* [34]. Valgren's work is similar to ours in the sense that it uses image features to localise a robot in a changing, but explored environment. Our approach uses global features instead of local ones. It deals with the characteristics of flying robots and uses solely the built-in camera of a mobile phone instead of an omni-directional camera.
Some UAV systems perform complex vision-based applications not onboard, but on a ground station [7], because they demand high computational power. We decided not to use a ground station, but to perform the entire image processing onboard a mobile phone. Although, the technical limitations of mobile devices make it difficult to implement image processing on them, there are approaches in context of user interaction and information services. Wang *et al.* [31] detect the motion of a mobile phone by analysing image sequences with motion estimation. While Castle *et al.* [5] compute visual SLAM for wearable cameras on a 3.2 GHz Pentium 4 CPU, there are image processing methods which run on a Nokia N95 mobile phone. Wagner *et al.* [30] implemented the *Scale Invariant Feature Transform (SIFT)* [13] on a N95 to find known textured targets. The mobile infomation system by Takacs *et al.* [26] uses CBIR. Here, users take pictures of a building and the system finds similar images in a database by comparing *Speeded Up Robust Features (SURF)* [3]. The system further provides the user with information about the building. In place of local image features, we study the applicability of global features for vision-only localisation.

3 SYSTEM COMPONENTS

The main components of our robot system are the quadrocopter and a mobile phone (Fig. 1). A quadrocopter is a fixed-wing aircraft which is lifted and propelled by four rotors. It is controllable by variation of the rotor's rotation speed. The lightweight quadrocopter is a commercial Asctec X3D-BL model quadrocopter, which is in detail described in [8]. It is equipped with a GPS sensor which reaches an accuracy of about 2.5 m. The quadrocopter lifts a maximum payload of 300 g. Thus, we need a lightweight device which supports a camera, computing power, memory, and which is able to connect to the quadrocopter. Because of these requirements we assembled a mobile phone, which supports all these functionalities, under the quadrocopter.
We use the N95 by Nokia with the operating system Symbian OS. Phones are embedded systems with limitations both in the computational facilities (low throughput, no floating point support) and memory bandwith (limited storage, slow memory, tiny caches). The N95 is based on an Arm 11 dual processor at 332 MHz. It contains 160 MB internal memory and an 8 GB memory card. Furthermore, the mobile device includes a 5-megapixel camera, a GPS sensor and data transfer techniques such as WLAN, in-

Fig. 1 The quadrocopter Asctec X3D-BL with a Nokia N95 mobile phone.

frared and Bluetooth. To apply these technologies, the mobile phone is programmable in Symbian C++, which offers a fast method to read the camera viewfinder images (sized 320×240 pixels, QVGA) in 3 ms.
The images, taken during flight missions, are stored in an image database for later monitoring. The image features are stored in a relational database system. Symbian OS supports a database API in the database language SQL.

4 LOCALISATION PROCESS

The localisation consists of two steps, an *exploration phase* and the actual *localisation phase* (Fig. 2). During the exploration phase, aerial images are taken. We extract characteristic features from these images and store them in a database. During the localisation phase, the quadrocopter navigates in an already known area. The mobile phone takes pictures and extracts their features. We compare these features to those from the database. Similar features and therewith similar looking images are supposed to originate from the same location.

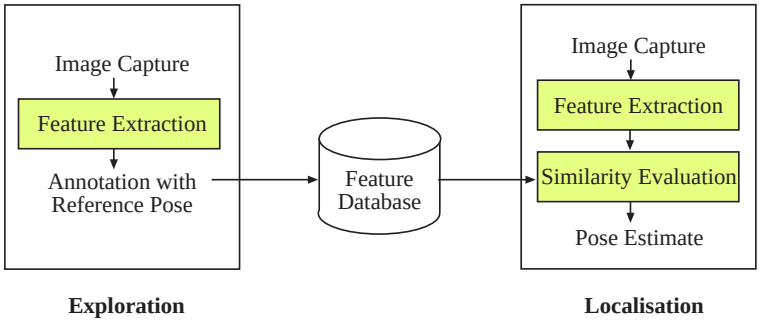


Fig. 2 The process consists of an exploration phase and the actual localisation phase. The highlighted steps are under investigation in this paper.

(a) Illumination changes

(b) Rotation

(c) Translation

(d) Rotation, translation and illumination changes

Fig. 3 Images of a building under usual transformations

4.1 Exploration Phase

The goal of the exploration phase is to map unknown areas in which the robot should be localised later. On exploration missions, the camera takes pictures of size 320×240 pixels and annotates them with their recorded positions. We used GPS as ground truth, because the values did not significantly deviate from the real position, as it can be seen in Fig. 6. At this stage, the mobile application requests the GPS data from the quadrocopter via infrared connection. We decided not to use the mobile phone's GPS sensor, because it is restricted by a fee required signing process. The GPS information is stored in the *Exif*[1]-header of each captured *JPEG*-image. For the future, we plan to

[1] Exchangeable Image File Format

turn our mapping stage into a SLAM system in order to eliminate the depending on GPS. In this stage, however, we used GPS as ground truth in order to benchmark our approach.
During flight, the mobile application extracts features which describe image characteristics. Compared to robots on the ground, aerial systems differ in their movement in six DoF $(\theta, \phi, \psi, x, y, z)$. Image features have to be stable to changes of illumination and they have to cope with image transformations in six DoF. The camera performs large rotation and translation motions during flight, which can be seen in Fig. 3. It shows that images taken at the same location do not have to be similar. We examine to what extent global image features are capable of encoding image similarities. We expect that the localisation accuracy will suffer from intense image transformations, but that global image features should already lead to robust localisation. This limitation is compensated by the relative simplicity of the approach. Further, we expect that the features are fast to compute and need little memory, which enables the mobile computation.
In the following, we introduce the three global feature extraction algorithms.

4.1.1 Grid Greyscale Histogram

A straightforward approach to characterize an image as a whole is to build a greyscale histogram. Ulrich and Nourbakhsh [28] established visual self-localisation for topological place recognition using color histograms. We extended the approach on two points. First, the illumination of the environment affects greyscale histograms. To obtain robustness to illumination changes, the grey value scale is divided into eight parts. Thus, one histogram bin represents 32 grey values. And second, partial occlusion has less impact on the histogram if the image is divided into subimages. By this means, changes of a subimage do not influence the whole histogram, but only the part of the histogram which represents the affected subimage. Therefore, we split the image into 4×4 subimages. For each subimage, we then compute a separate histogram. Translations or rotations of the image cause pixels to fall into another subimage. Pixels in the centre of a subimage obtain a larger weight than those near the subimage border. The weighting is realized by a Gaussian function placed in the centre of each subimage.
To get the final 1×128 feature vector, the histograms of the subimages are concatenated. The vector is normalized according to the chosen norm (Sect. 4.2).
We expect the algorithm to be sensitive to changing lighting conditions, but at the same time, it is fast to compute.

4.1.2 Weighted Grid Orientation Histogram

Bradley *et al.* [4] presented the *Weighted Grid Orientation Histogram* (WGOH) approach in 2005. It is inspired by the popular SIFT by Lowe [13]. The feature uses weighted histograms of image gradient orientations. In contrast to the local SIFT feature, histograms are not computed around interest points, but globally for each pixel. We chose WGOH, because it was reported to obtain good results under different illumination conditions [4]. It benefits from the robustness of the SIFT idea and the simplicity of the global approach.
The algorithm works as follows (Fig. 4): The image is divided again into 4×4 subimages. For each pixel the orientation of the gradient is calculated. A histogram of gradient orientations is built for each subimage. For this purpose, the orientations are divided

into eight intervals. The gradient orientations are weighted by the magnitude of the gradient. Additionally, a Gaussian function is used for weighting each magnitude. The gradient histograms are concatenated to a 1×128 vector, which is normalized subsequently.

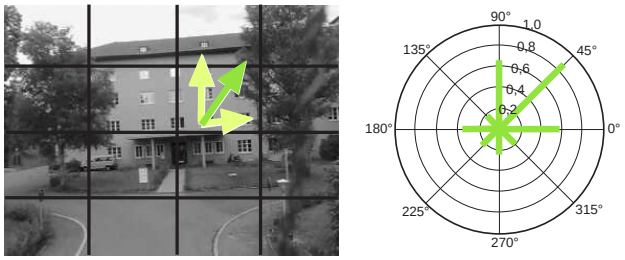


Fig. 4 Creation of the WGOH feature vector by determing the gradient orientation in each pixel and building an orientation histogram.

4.1.3 Weighted Grid Integral Invariants

The *Weighted Grid Integral Invariants* (WGII) approach by Weiss *et al.* [32] is based on Integral Invariants, which were first introduced by Manay *et al.* [15]. A detailed description of Integral Invariants can be found in [22]. Weiss *et al.* achieved good results under illumination changes. The idea is to apply as many transformations as possible to an image and to integrate the outcomes. Integration makes the features independent of the sequence of the transformations and leads to invariance against these transformations.

Integral Invariants proceed as follows (Fig. 5): A kernel function f involves the local

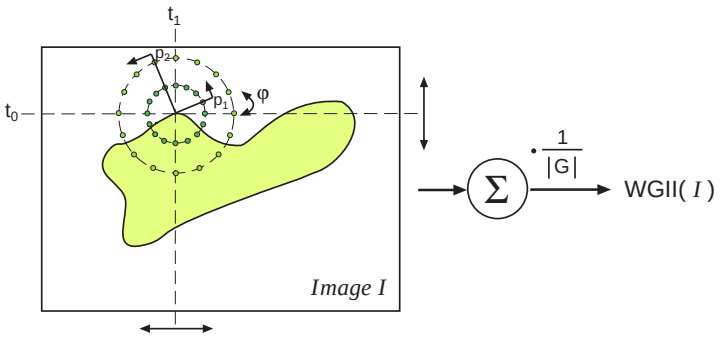


Fig. 5 The creation of the WGII feature vector by rotating a pair of neighbor pixels and averaging over the difference of their intensities.

neighborhood of each pixel. The function subtracts the intensities of two pixels lying

on circles with different radii around the considered pixel. The two observed pixel positions are rotated 10 times. The kernel function is evaluated for each rotation g. Then, we integrate over these evaluations to achieve independence of the order of transformations. The Integral Invariant feature of the image $\mathbf{I}$ of size $N_0 \times N_1$ is calculated by

$$F(\mathbf{I}) = \frac{1}{RN_0N_1} \sum_{t_0=0}^{N_0-1} \sum_{t_1=0}^{N_1-1} \sum_{r=0}^{R-1} f\Big(g\big(t_0, t_1, 2\pi\frac{r}{R}\big)\mathbf{I}\Big)\,, \tag{1}$$

where R is the number of rotations. Integral Invariants are applied to each pixel and weighted by a Gaussian function. The image is split in 4×4 subimages. To calculate the Integral Invariants, we use two pairs of pixels $(2;0)^T$, $(0;3)^T$ and $(5;0)^T$, $(0;10)^T$ as kernels. The final feature is a histogram of the Integral Invariants. We use an 8-bin histogram per subimage. These histograms of sixteen subimages and both kernels are concatenated to obtain the final 1×256 feature vector.

4.2 Localisation phase

During the localisation phase, the quadrocopter flies in an already visited area. The mobile application takes pictures and compares them with those in the database. To compare two pictures, it is not only important to find significant features, but to define a similarity measure to detect similar features. In the following, we define five measures which compare histograms, represented by vectors.
In [22], Siggelkow distinguished two concepts to compare feature histograms: a *bin-by-bin measure* and a *cross-bin comparison measure*. In the work at hand, we use bin-by-bin measures, which compare corresponding bins. The reason for this is that bin-by-bin measures have linear complexity in the number of bins, while cross-bin comparison measures have higher complexity. The notation of histograms corresponds to notation of vectors $h = (h_0, h_1, ..., h_{M-1})^T$.
First, we remember the well-known Minkowski distance. It is defined by

$$d_{L_p}(h^{(0)}, h^{(1)}) = \left(\sum_{m=0}^{M-1} |h_m^{(0)} - h_m^{(1)}|^p\right)^{\frac{1}{p}}. \tag{2}$$

We consider three choices of p: the L_1*-norm* or *Manhattan norm* distance with $p = 1$, the *Euclidian distance* with $p = 2$ and the *infinity norm* distance with $p \to \infty$:

$$\begin{aligned} d_{L_\infty}(h^{(0)}, h^{(1)}) &= \lim_{p\to\infty} \left(\sum_{m=0}^{M-1} |h_m^{(0)} - h_m^{(1)}|^p\right)^{\frac{1}{p}} \\ &= \max_{m=0,...,M-1} \left(|h_m^{(0)} - h_m^{(1)}|\right). \end{aligned} \tag{3}$$

The infinity norm performs bin-by-bin comparisons and considers the largest difference as the measurement value. If a single pair of bins is very different and all other bins are similar to each other, the pair of feature histograms is not regarded as a match.

Furthermore, Siggelkow introduced another distance with intuitive motivation. It is called the *intersection measure* $d_\cap$ and calculates the common part of both histograms:

$$d_\cap \left(h^{(0)}, h^{(1)} \right) = \sum_{m=0}^{M-1} \min \left(h_m^{(0)}, h_m^{(1)} \right). \tag{4}$$

Bradley *et al.* [4] use the following measure to compare two normalized WGOH feature vectors:

$$d_{Bradley} \left(h^{(0)}, h^{(1)} \right) = 1 - \left(h^{(0)} \right)^T h^{(1)}. \tag{5}$$

It is based on the fact, that the scalar product of two identical vectors is 1. The reverse does not hold: If a scalar product of two vectors is 1, the vectors do not have to be identical.
The intersection measure and Bradley's measure are not induced by norms. Thus, we normalized the feature vectors depending to the norm, which is the most suitable to the measures. We chose the L_1-norm for the intersection measure and the Euclidian norm for Bradley's measure, because experimentally, these combinations yielded the best results.
In the following we examine to what extent the measures influence the image feature approaches.

5 EXPERIMENTAL RESULTS

In this section, we analyse the presented localisation system on the basis of two datasets and different configurations of the feature algorithms and similarity measures.

5.1 Datasets

The monocular camera of our aerial system faces forwards. Thus, a quadrocopter which rotates or varies its altitude, has many different views at the same latitude and longitude position. Besides, a large amount of data has to be recorded. To organize the exploration phase effectively and to minimize the database, we restrict our setting: According to the quadrocopter's task, the altitude and orientation are constrained. We collect data in an exemplary task, which is a flight from a start to an end point for a given route. The view of the camera pointed in course direction. The quadrocopter's altitude varied between five and ten meters, which is the range of the desired altitude. The images of the dataset were taken at different illumination conditions.
The dataset consists of 1,275 images in total, taken during a traversal of flying eight rounds in a courtyard. During an exploration mission, we took 348 training images of the courtyard with diffuse illumination (dataset T), representing one round. Dataset T was chosen to serve as our reference dataset, because it was a slow-moving exploration flight with twice the number of images as the test flights (see Table 1). Dataset C consists of 558 images taken at four rounds at cloudy weather. Dataset S contains 369 images, representing three rounds, which were taken on a sunny day. The images show hard shadows (Fig. 3(a), left). The distance between images varies between the datasets. Fig. 6 shows one round of dataset C and S, respectively.

Table 1 Characteristics of the datasets

Dataset	Illumination	Number of images		Purpose
T	cloudy	348		Exploration
C1	cloudy	88	558	
C2	cloudy	125		
C3	cloudy	142		
C4	cloudy	203		Localisation
S1	sunny	94	369	
S2	sunny	118		
S3	sunny	157		

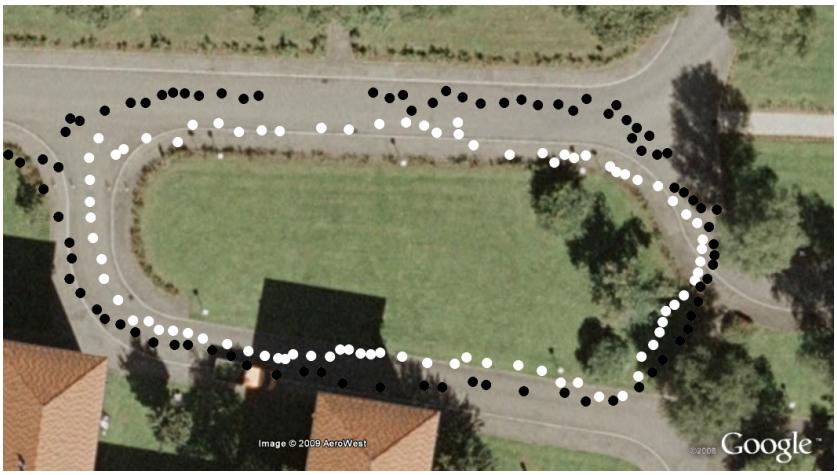


Fig. 6 The localisation area: The route is about 180 meters in length. The captured images are placed in the aerial picture by their geotags. The two rounds can be clearly distinguished from each other (dataset C1 (white); dataset S1 (black)). GPS values are consistent and not falsified by erroneous sognals. The distances between images vary because of gusts of wind.

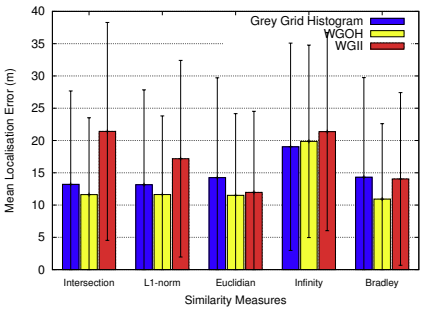
5.2 Results

In the following analysis, we evaluate the matching results of the visual localisation algorithms and the corresponding measures. GPS data were added to all datasets, serving as ground truth positions.

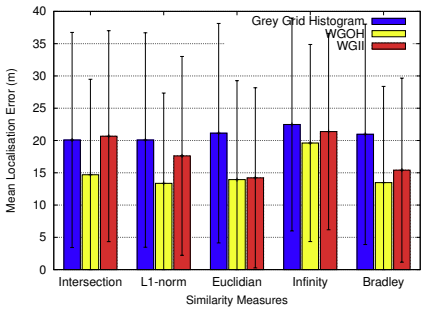
5.2.1 Comparison of Similarity Measures

In this section we analyse the matching accuracy for the different similarity measures. We measured the 2D position distance between the query image and the matched image for each feature algorithm and for each similarity measure.

The results of the different measures under varying illumination are shown in Fig. 7. The mean localisation error lies between 10.9 m and 21.4 m. Considering this error, the appropriate choice of image feature and distance measure is important. With regard to the intense image transformations and the size of the test area of about 180 meters



(a) Similar illumination during exploration and localisation phase.



(b) Different illumination conditions in the two phases.

Fig. 7 Comparison of the similarity measures combined with the different features. 7(a) shows the mean absolute localisation error and standard deviations under comparable illumination conditions during exploration and localisation (dataset C), while 7(b) is evaluated under different conditions (dataset S).

in length, we consider the localisation error as good. An image taken close to a query image can look very different. However, an image which is taken from a 20 m distance can picture the same motif (Fig. 8).

Figure 7(a) shows the results under similar illumination of training and test images.

Fig. 8 Combination (right) of a query (left) and a matched image (middle) which have a distance of 20 m to each other and which are congruent to nearly a third.

It illustrates the importance of the correct choice of distance metric for the particular feature extraction algorithm. The localisation error with the WGII approach varies about 10 m, depending on the chosen measure. In contrast, WGOH is only marginal influenced by the chosen measure. The measurements with the Euclidian norm and the Bradley measure (Equ. 5) perform with similar good results on all features. Each feature requires a different measure to yield the best performance. WGOH convinces in conjuction with Bradley's norm resulting in a localisation error of 10.92 m. WGII achieves best results (11.5 m) with the Euclidian norm. The grid greyscale histogram works best with the intersection measure (13.2 m) and the L_1-norm (13.15 m). The results of the infinity norm are worst. Here, only histograms are matched whose bins are without exception similar to its correspondent bins. This approach is not robust enough for our aerial images.

The experiments with dataset S under different illumination conditions are shown in Fig. 7(b). The localisation error increases by 3 m on average. This difference is not

caused by the measures, but by the feature extraction algorithms which are analysed in the following.

5.2.2 Comparison of Feature Extraction Algorithms

Having found the most appropriate similarity measure for each approach, we concentrate on comparing the feature extraction algorithms.
As shown in Fig. 7, WGOH achieves the best results, followed by WGII. WGOH represents an image by its gradient orientations, while WGII examines the differences of neighbouring pixel intensities. Both approaches can cope with intense image transformations like rotation and translation to a certain extent. WGOH divides gradient orientations into eight intervals, which means that the feature is rotation-invariant up to 45 degrees. WGII rotates its kernels ten times. Thus, it is invariant to 36°-rotations. Both features divide the image into 4×4 subimages. Grid building helps being robust to occlusions.
Concerning the good results of the Euclidian norm, we restricted the following analysis of the influence of illumination changes by using the Euclidian norm (Fig. 9). WGOH yields worse results under heterogenous illumination during training and testing phase of 2.44 m. The localisation error of WGII increases by 2.26 m, and the one of the greyscale histogram by 6.88 m. WGOH makes use of gradients. Images taken at sunshine show deep shadows with gradients large in magnitude which exert influence on the WGOH feature. WGII is least affected by illumination changes. It compares only the difference between neighbouring pixel intensities. Varying illumination has only little effect on the difference in intensity. For instance, plain-coloured areas likely remain plain-coloured despite illumination variations. The grid greyscale histogram shows surprisingly good results under similar lighting conditions. In conjuction with the intersection and L_1-norm, the greyscale histogram works better than WGII. But it suffers under different illumination conditions, because it is solely based on greyscale values.

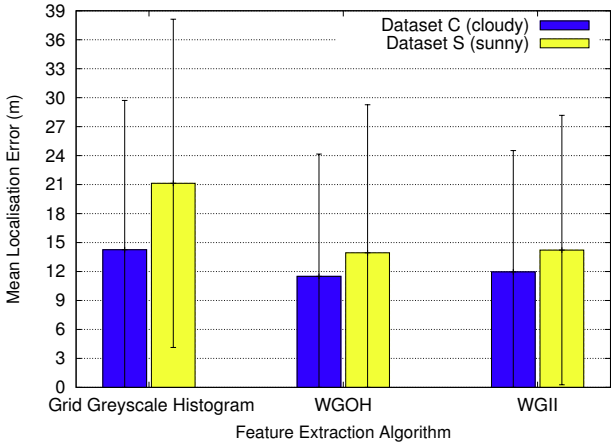


Fig. 9 Comparison of the different feature algorithms under the Euclidian norm and varying illumination conditions.

Table 2 Computing time [s]

		WGOH	WGII	Histogram
a	Feature Calculation	0.49	4.39	0.16
b	Database Creation	0.66	0.66	0.66
c	Single feature comparison to a database of 348 entries	0.15	1.17	0.15
a+b	Exploration	1.15	5.05	0.82
a+c	Localisation	0.64	5.56	0.31

5.2.3 Computation Times

Table 2 shows the computation time required by the different algorithms for the Euclidian measure. The greyscale histogram is the fastest approach, followed by WGOH and finally WGII. During localisation phase, the histogram needs 0.31 s and WGOH needs 0.64 s to process an image. WGII requires about 5.5 s to compute and match features, and hence more than eight times as long as WGOH. This can be improved by calculating Integral Invariants in every forth randomized pixel location in the image. In this case, the mean localisation error worsens from 11.96 m to 12.83 m, but the time consumption of the feature extraction reduces from 4.4 s to 1.3 s.

6 CONCLUSIONS

In this research, we visually localised a small UAV with techniques applied onboard in a large (180 m) outdoor environment. We used only a mobile phone as visual sensor and onboard vision processing computer. We tested the system under image transformations and illumination changes with different algorithm and similarity measue configurations. The feature extraction algorithm WGOH and the feature comparison measure Euclidian norm worked best. The computing time in the exploration phase is 1.15 s, the localisation phase takes 0.64 s. The mean localisation error is 10.92 m. This is comparable to Valgren's [29] results in large outdoor environments. Valgren *et al.* defined a threshold at a distance of 10 meters between match and query image to classify a match as correct.

We see great potential in using our system in real-world applications. Once an environment has been explored with accurate GPS position values, the system does not depend on GPS anymore. If GPS is unavailable or noisy, we can switch to the visual localisation system. We will concentrate on a SLAM system to eliminate the GPS-dependent exploration phase.

In future work we focus on the acceleration of the localisation process. Our prototype implementation can be improved by integer optimized algorithms and SIMD instructions.

Further work includes expanding the experiments to a larger environment. In addition, a particle filter will be used to improve the localisation error.

References

1. Artac, M., Leonardis, A.: Outdoor mobile robot localisation using global and local features. In: Proceedings of the 9th Computer Vision Winter Workshop (CVWW), pp. 175–184. Piran, Slovenia (2004)
2. Bath, W., Paxman, J.: UAV localisation and control through computer vision. In: Proceedings of the Australasian Conference on Robotics and Automation. Sydney, Australia (2004)
3. Bay, H., Ess, A., Tuytelaars, T., Gool, L.: Surf: Speeded up robust features. In: Computer Vision and Image Understanding (CVIU), vol. 110, pp. 346–359 (2008)
4. Bradley, D., Patel, R., Vandapel, N., Thayer, S.: Real-time image-based topological localization in large outdoor environments. In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS), pp. 3670–3677. Edmonton, Alberta, Canada (2005)
5. Castle, R., Gawley, D., Klein, G., Murray, D.: Towards simultaneous recognition, localization and mapping for hand-held and wearable cameras. In: Proceedings of the International Conference on Robotics and Automation (ICRA), pp. 4102–4107. Rome, Italy (2007)
6. Corke, P., Strelow, D., Singh, S.: Omnidirectional visual odometry for a planetary rover. In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS), vol. 4, pp. 4007–4012. Sendai, Japan (2004)
7. De Wagter, C., Proctor, A., Johnson, E.: Vision-only aircraft flight control. In: The 22nd Digital Avionics Systems Conference (DASC), vol. 2, pp. 8.B.2–81–11 (2003)
8. Gurdan, D., Stumpf, J., Achtelik, M., Doth, K.M., Hirzinger, G., Rus, D.: Energy-efficient autonomous four-rotor flying robot controlled at 1 khz. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp. 361–366. Roma, Italy (2007)
9. Hofmeister, M., Liebsch, M., Zell, A.: Visual self-localization for small mobile robots with weighted gradient orientation histograms. In: 40th International Symposium on Robotics (ISR), pp. 87–91. Barcelona, Spain (2009)
10. Jogan, M., Leonardis, A., Wildenauer, H., Bischof, H.: Mobile robot localization under varying illumination. In: 16th International Conference on Pattern Recognition (ICPR), vol. II, pp. 741–744. Los Alamitos, CA, USA (2002)
11. Kim, J., Sukkarieh, S.: Real-time implementation of airborne inertial-SLAM. In: Robotics and Autonomous Systems archieve, vol. 55, pp. 62–71. Amsterdam, The Netherlands (2007)
12. Lamon, P., Nourbakhsh, I., Jensen, B., Siegwart, R.: Deriving and matching image fingerprint sequences for mobile robot localization. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp. 1609–1614. Seoul, Korea (2001)
13. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of the International Conference on Computer Vision (ICCV), pp. 1150–1157. Corfu (1999)
14. Maimone, M., Cheng, Y., Matthies, L.: Two years of visual odometry on the mars exploration rovers: Field reports. Journal of Field Robotics **24**(3), 169–186 (2007)
15. Manay, S., Hong, B.W., Yezzi, A.: Integral invariant signatures. In: Proceedings of the 8th European Conference on Computer Vision (ECCV), vol. LNCS 2034, pp. 87–99. Prague, Czech Republic (2004)

16. Mondragon, I., Campoy, P., Correa, J., Mejias, L.: Visual model feature tracking for UAV control. In: IEEE International Symposium on Intelligent Signal Processing (WISP), pp. 1–6. Alcala de Henares (2007)
17. Musial, M., Brandenburg, U., Hommel, G.: Cooperative autonomous mission planning and execution for the flying robot MARVIN. In: Intelligent Autonomous Systems 6, pp. 636–643. Amsterdam, The Netherlands (2000)
18. Pinies, P., Lupton, T., Sukkarieh, S., Tardos, J.D.: Inertial aiding of inverse depth SLAM using a monocular camera. In: Proceedings IEEE International Conference on Robotics and Automation (ICRA), pp. 2797–2802. Rome, Italy (2007)
19. Saripalli, S., Montgomery, J.F., Sukhatme, G.S.: Vision-based autonomous landing of an unmanned aerial vehicle. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 2799–2804. Washington, DC, USA (2002)
20. Scaramuzza, D., Siegwart, R.: Appearance guided monocular omnidirectional visual odometry for outdoor ground vehicles. IEEE Transactions on Robotics, Special Issue on Visual SLAM, vol. 24, issue 5 (2008)
21. Scaramuzza, D., Siegwart, R.: Monocular omnidirectional visual odometry for outdoor ground vehicles, vol. 5008/2008, chap. Computer Vision Systems, Lecture Notes in Computer Science, pp. 206–215. Springer Press (2008)
22. Siggelkow, S.: Feature histograms for content-based image retrieval. Ph.D. thesis, Albert-Ludwigs-Universität Freiburg, Fakultät für Angewandte Wissenschaften, Freiburg, Germany (2002)
23. Sim, R., Dudek, G.: Comparing image-based localization methods. In: Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI), pp. 1560–1562. Apaculco, Mexico (2003)
24. Sinopoli, B., Micheli, M., Donato, G., Koo, T.J.: Vision based navigation for an unmanned aerial vehicle. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA), pp. 1757–1765. Seoul, Korea (2001)
25. Steder, B., Rottmann, A., Grisetti, G., Stachniss, C., Burgard, W.: Autonomous navigation for small flying vehicles. In: Workshop on Micro Aerial Vehicles at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). San Diego, CA, USA (2007)
26. Takacs, G., Chandrasekhar, V., Gelfand, N., Xiong, Y., Chen, W., Bismpigiannis, T., Grzeszczuk, R., Pulli, K., Girod, B.: Outdoors augmented reality on mobile phone using loxel-based visual feature organization. In: Proceedings of the 1st ACM international conference on Multimedia information retrieval (MIR), pp. 427–434. New York, NY, USA (2008)
27. Tamimi, H.: Vision-based features for mobile robot localization. Ph.D. thesis, Eberhard-Karls-Universität Tübingen, Tübingen, Germany (2006)
28. Ulrich, I., Nourbakhsh, I.: Appearance-based place recognition for topological localization. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA), pp. 1023–1029. San Francisco, CA, USA (2000)
29. Valgren, C., Lilienthal, A.: SIFT, SURF and seasons: Long-term outdoor localization using local features. In: Proceedings of the European Conference on Mobile Robots (ECMR), pp. 253–258. Freiburg, Germany (2007)
30. Wagner, D., Reitmayr, G., Mulloni, A., Drummond, T., Schmalstieg, D.: Pose tracking from natural features on mobile phones. In: 7th IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR), pp. 125–134. Cambridge, UK (2008)

31. Wang, J., Zhai, S., Canny, J.: Camera phone based motion sensing: interaction techniques, applications and performance study. In: Proceedings of the 19th annual ACM symposium on user Interface software and technology (UIST), pp. 101–110. New York, NY, USA (2006)
32. Weiss, C., Masselli, A., Tamimi, H., Zell, A.: Fast outdoor robot localization using integral invariants. In: Proceedings of the 5th International Conference on Computer Vision Systems (ICVS). Bielefeld, Germany (2007)
33. Williams, B., Klein, G., Reid, I.: Real-time SLAM relocalisation. In: IEEE 11th International Conference on Computer Vision (ICCV), pp. 1–8. Rio de Janeiro, Brazil (2007)
34. Zhou, C., Wei, Y., Tan, T.: Mobile robot self-localization based on global visual appearance based features. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp. 1271–1276. Taipei, Taiwan (2003)