Information Overlay for Camera Phones in Indoor Environments

Harlan Hile and Gaetano Borriello

Dept. of Computer Science and Engineering
University of Washington
Box 352350
Seattle, WA 98195-2350 USA
{harlan,gaetano}@cs.washington.edu

Abstract. Increasingly, cell phones are used to browse for information while location systems assist in gathering information that is most appropriate to the user's current location. We seek to take this one step further and actually overlay information on to the physical world using the cell phone's camera and thereby minimize a user's cognitive effort. This "magic lens" approach has many applications of which we are exploring two: indoor building navigation and dynamic directory assistance. In essence, we match "landmarks" identified in the camera image with those stored in a building database. We use two different types of features – floor corners that can be matched against a floorplan and SIFT features that can be matched to a database constructed from other images. The camera's pose can be determined exactly from a match and information can be properly aligned so that it can overlay directly onto the phone's image display. In this paper, we present early results that demonstrate it is possible to realize this capability for a variety of indoor environments. Latency is shown to already be reasonable and likely to be improved by further optimizations. Our goal is to further explore the computational tradeoff between the server and phone client so as to achieve an acceptable latency of a few seconds.

1 Introduction

As someone walks through a building, he may want help navigating to the right room or accessing dynamic directory information related to his current location. Both of these tasks would benefit from the environment directly providing personalized information. Current approaches to this problem use location systems to help the user index into the information. However, the retrieved information must be put into context. We use a camera phone to provide the contextual framework by having the act of pointing the camera form the query into the information. The phone's display can then be used to overlay information directly onto the image and provide the user information in context. The crucial element of this approach is in determining the precise camera pose (location in 3D and orientation - a full 6 degrees of freedom).

J. Hightower, B. Schiele, and T. Strang (Eds.): LoCA 2007, LNCS 4718, pp. 68–84, 2007.

We are working on two applications of this capability. The first is motivated by an ongoing project to help individuals with cognitive impairements navigate in indoor spaces. This user population often has difficulty navigating complex buildings such as medical centers and shopping malls. In essence, we are trying to provide customized "painted lines on the floor" for users to follow to their destination. Our goal is to overlay directional arrows and navigation instructions onto the image as it is easier to understand directions when they are overlaid directly on the user's own view of his environment, especially for people with cognitive impairments [1]. The inability to get around efficiently can limit integration into the community or affect their ability to be gainfully employed. We are building a system that supports both indoor and outdoor navigation; here we focus only on the indoor portion. The second application targets a more general population that may be interested in finding out information about a building such as what events are taking place, which resources are reserved and by whom, and when someone was last in their office. This is what we mean by "dynamic directory information". As people walk down hallways, they should be able to see customized "dynamic name plates" that provide this data. Examples of information overlay for both of these uses are shown in Figure 1.

Our approach to finding the camera pose is based on a simple concept: we determine the "landmarks" in the image and their correspondence to previously cached landmarks of the space. By matching enough landmarks we can precisely compute the camera pose and thereby accurately overlay information onto the display. This simple idea is complicated by the fact that different spaces have different types of landmarks. We use "micro-landmarks" such as floor/wall transitions, corners, and door/floor edges when we are in spaces that have a high degree of homogeneity, such as hallways. We use "texture-landmarks" in larger and richer spaces such as open areas (e.g., an atrium or large room).

In the case of hallways, we can determine a user's location on the floorplan of a building by comparing the camera image to the floorplan. Our image processing targets micro-landmarks likely to be found on the floorplan, e.g., position of doors, hallway intersections, etc. We compare those found in the image to a floorplan provided by the building's infrastructure. We use a building server to hold this floorplan data as well as provide the computation cycles for extracting the micro-landmarks and performing the matching. Communication between the client and server is realized through a Wi-Fi connection supported by many newer phone models. We prune the search of the best correspondence by using the Wi-Fi fingerprint to coarsely locate the user (we only expect a location estimate that is accurate to within 5-10 meters - easily attainable with several of today's Wi-Fi-based positioning systems [2,3]).

The problem of determining useful landmarks is complicated in large open spaces as it is much more difficult to discern floorplan features. In this case, we use an approach developed by Snavely, Seitz, and Szeliski named Photo Tourism [4] which uses texture features of the images computed by SIFT [5] which are mapped to 3D locations through the use of multiple images. This stored structure allows a similar correspondence to be computed between the landmark database

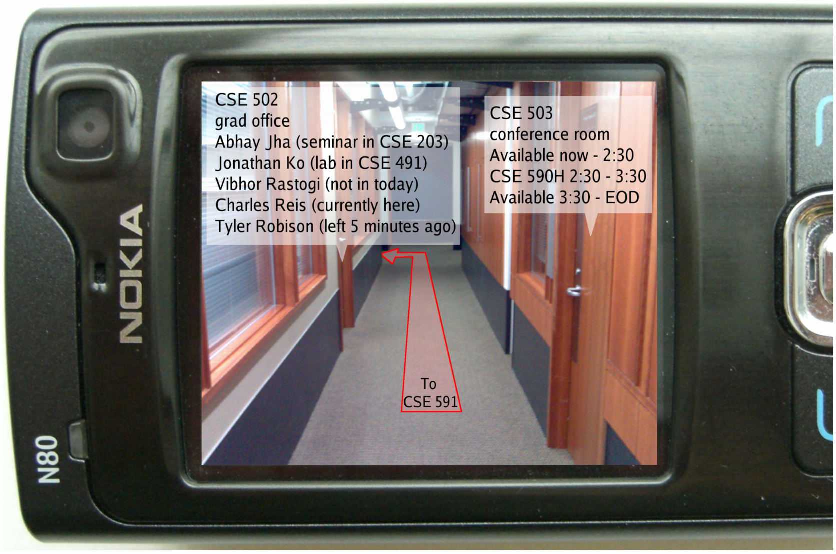

Fig. 1. Sample of possible information overlay on an image of the environment. Image shows both an overlaid navigation aid and a magic lens-type application with dynamic information about the current surroundings.

and new images. These texture-landmarks do not rely on specific structural elements. It is important, however, to ensure that the landmark database used for feature correspondence is an accurate reflection of the current space.

Thus, our system works as follows: (0) the phone captures the image, (1) sends it to the server along with Wi-Fi fingerprints and the type of information requested, (2) the server performs feature extraction and (3) finds the correspondence to the building's floorplan or to previously captured images, (4) from this correspondence the camera pose can be computed, and finally, (5) the server returns an information overlay for display to the phone client. Figure 2 provides a diagram of how our system operates (focusing on the micro-landmarks used for hallways). Our challenge is to ensure that all this computation and communication can be performed in under 5 seconds to support reasonable user interaction speeds. Although we currently do all the processing on the building's server our goal is to explore different partitions of tasks and evaluate their performance. Moreover, we can further explore the types of features/landmarks we extract from the image and consider several hybrid approaches. We also plan to investigate the use of video (as the user moves the phone) as this may provide more cues than a static image. Video processing is likely to be better suited for this type of "magic lens" or augmented reality application. We expect that the current single image technique can be extended with refinement techniques to support video.

In the following sections of this paper, we discuss work on similar problems and how they relate to our system. Then we examine the steps of processing an

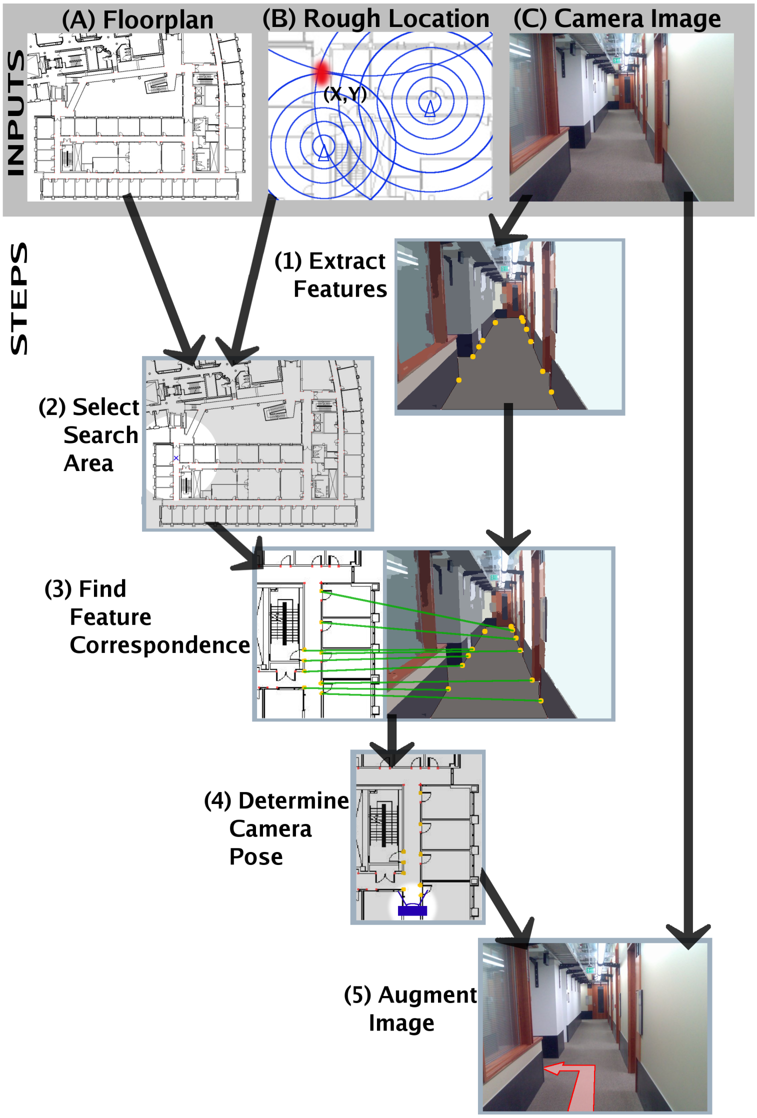

Fig. 2. System diagram for calculating camera pose and overlaying information on an image from a camera phone. The inputs to the system are the following: (A) a floorplan with relevant features marked on it, (B) a rough location estimate (which includes floor information), and (C) an image from the camera. Using this information, the first step is to extract relevant features from the image. Step two is to choose a region of the floorplan and a set of features to match against. In step three, a mapping between the features in the image and features on the floorplan must be assigned and evaluated. In step four, this information can then be used to find a more precise location and orientation on the floorplan. Lastly in step five, location of objects and features on the floorplan can then be transferred onto the image, along with data relevant to the user. A similar methodology exists for texture landmarks.

image and matching it to a floorplan, illustrated by examples. Lastly we discuss remaining issues and extensions to our system.

2 Related Work

A variety of systems exist that can localize a device in its environment with the use of an image. Many of these are designed for robot navigation, and make use of odometry data and rely on many images to localize accurately. Early work shows that matching building features to refine location is feasible [6], but requires a highly accurate estimate of current position and orientation. More recent work makes use of robust image features such as those produced by SIFT [5]. The vSLAM system simultaneously builds a map and localizes images features within that map [7]. Other robotic navigation systems use many images along a path and visibility information to calculate likely locations [8,9]. None of these systems are suitable for our desired scenario where odometry data and multiple images are not available.

Image analysis systems that are not intended for localization also provide useful components. Photo Tourism can solve for 3-D locations of feature points in a scene and calculate camera positions from a large group of images [4]. New images can be matched to the current model, producing an accurate camera pose in relation to the scene. Annotations can be transferred between images by using the underlying 3-D structure. Photo Tourism relies on distinct SIFT features which most hallways lack, and it is not designed for quickly finding a camera position in a large area given a location estimate. For this reason, the approach of Photo Tourism plays an important role in our system, but is not sufficient. Systems to recognize landmarks in outdoor environments and provide annotations also exist [10]. This also relies on SIFT features, and does not actually generate a refined location, but merely identifies what objects might be in the image. Providing a database of geocoded features is also much higher cost than providing a building floorplan. Although these systems support similar interactions, they are not suitable for use on hallway images.

Augmented reality systems share a similar goal of information overlay. These systems tend to be object centric, and often tag objects with special markers to facilitate their location in an image [11,12,13]. Other systems actively project structured light on the environment to aide in localization [14], but would have difficulty in a hallway environment. Existing augmented reality systems provide a variety of examples of information overlay and may be a source for applications of this system, but do not currently support hallway environments without special tagging or special hardware. Although some systems seek to augment existing maps [15], our system aims to bring the map information to the user's current environment.

3 System

As previewed in Figure 2, the steps of the system will be first to extract possible relevant features from a given 640x480 cell phone camera image. Next a relevant section of the floorplan (or other feature database) must be chosen, and then the matching will be performed. Once the correspondence is found, it can be used to find location and orientation of the camera. Lastly, the location of objects and other relevant information can be transferred back into image space. The following sections will discuss these steps in more detail, illustrated by examples from work on the problem. Although the bulk of our current work focuses on hallway images and building features, we also discuss the use of image-based features that will support environments other than hallways.

3.1 Features in Homogeneous Environments (Hallways)

Feature Detection. The first step in finding how an image matches to a floorplan is to locate the features in the image. There is limited information on a standard floorplan, but any floorplan should include location of doorways and corners of walls. These features, or micro-landmarks, will be visible in the image too, and the goal is to find them. The concept is simple: to locate the lines that define the edge of the floor and the lines that define the edge of each doorway or corner. Intersecting these sets of lines will give points that correspond to the features on the floorplan.

We have implemented a basic feature detection method based on segmenting the image. Instead of looking for the edges of the floor directly, we locate the entire floor and then use the edges of that region. We have chosen the Mean Shift method to perform image segmentation [16]. The floor will then likely be the segment that is at the bottom center of the image. It is difficult to take a picture of a hallway where this is not the case, but this requirement could be included in the interface that prompts users to take pictures. The edge of this region is then traced and the corners are identified using a "cornerity" metric [17]. This will not locate all the places along the floor where there is a doorway, but it finds many of them. More can be located by intersecting vertical lines found in the image with the floor boundary, but this is not done in these examples. Additionally, this tracing method finds some false corners that do not correspond to anything in the floorplan. False corners that are at the top of a vertical line of the floor edge can be discarded as points likely caused by occlusion of the floor. The results of segmentation and corner finding are shown in Figure 3. These are points that can now be used to match to the floorplan.

Although this basic feature detection gives results suitable for demonstrating the system, we are looking at other methods to improve it. This will be discussed more in Section 4. Our current method is similar in spirit to other work done on natural indoor scene analysis for local robot navigation [18].

Feature Matching. Once the features in the image are found, they must be matched to the floorplan. The first step of this is to choose a set of points from the floorplan to match against. This is done both to remove ambiguous cases and to reduce the search space. The rough location estimate provides a center for the region to be tested, and a radius can be estimated based on the accuracy of the location system and some idea of the camera's useful visibility range. Figure 4

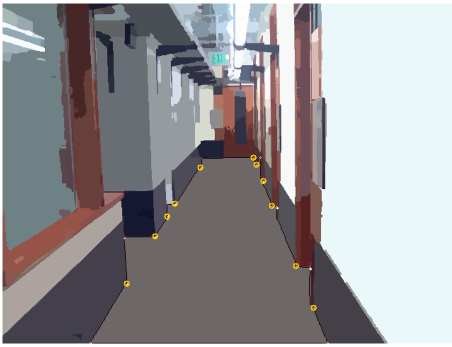

Fig. 3. The hallway image is first segmented with Mean Shift, then the edge of the floor traced and corners located. Corners marked with orange circles are candidates for matching to floorplan features.

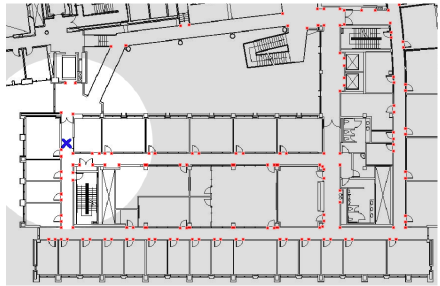

Fig. 4. The region of the floorplan considered for matching. The estimated location is marked with an X, and the feature points are marked with small red dots.

shows the floorplan, an estimate of the location, and the features that will be considered for matching for the hallway image shown in Figure 3. It is possible to include other information when calculating the region to consider, such as visibility calculations, or priors for direction, either from direction of motion or an external sensor, but we currently do not include these.

Once the two sets of features are defined, they must be matched up. This is a challenging problem because of the number of possible ways to match. A transformation between the image space and the map space can be defined by four sets of correspondences. For the examples presented here, there are an average of 10 image points and an average of 27 map points, resulting in over 2 billion possible four-point correspondences. In order to make this problem tractable, we perform this matching using a RANSAC approach that intelligently selects and prioritizes the hypotheses. The hypotheses are generated by randomly choosing two points from the image and two points from the map. These two points define a line in each space, and the next two points must have consistent placement with respect to that line across the image and map space. This selection eliminates approximately 90% of the hypotheses that would be generated purely randomly. Hypotheses are also prioritized based on the area covered by the four points in the image. The larger the distance between the points, the more likely it will produce a stable homography. The area of the bounding box of the four points is used as a measure of spread, and a threshold is slowly lowered in order to implement priority. Lastly, the search is terminated early if the estimate has not improved in a "long time" (for example, 5,000 samples) because it unlikely to improve further as the threshold lowers. Our examples obtain good results testing less than 100,000 hypotheses.

Hypotheses are evaluated by solving for a homography that maps from the image points to the floorplan points, and looking at the sum of squared distance between the two point sets. Weights for points closer to the camera (lower in the

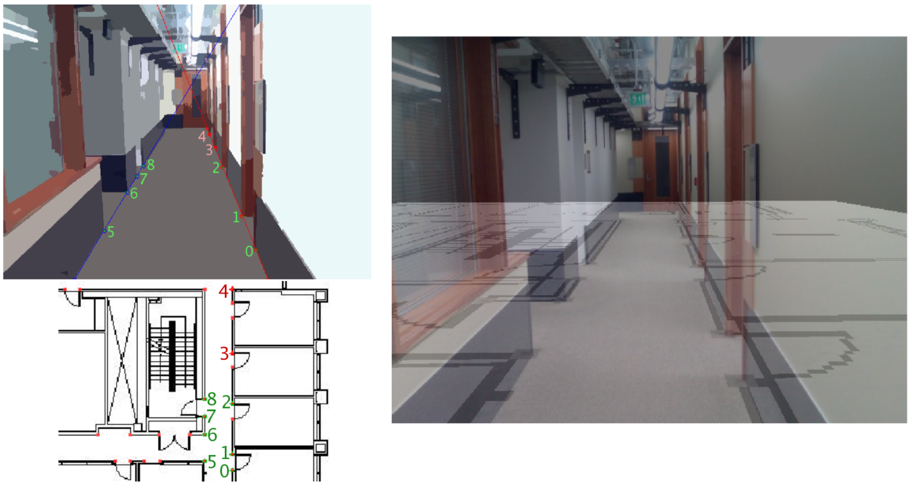

Fig. 5. Results of point correspondence algorithm. Upper left: the detected points on the image labeled by number. Lower left: the points on the floorplan that were matched to the image, labeled with corresponding numbers. Right: the floorplan warped into the image space and overlaid on top of the original image, which matches very well. This example has 10 image points and 32 map points, and completes matching in about 15 seconds.

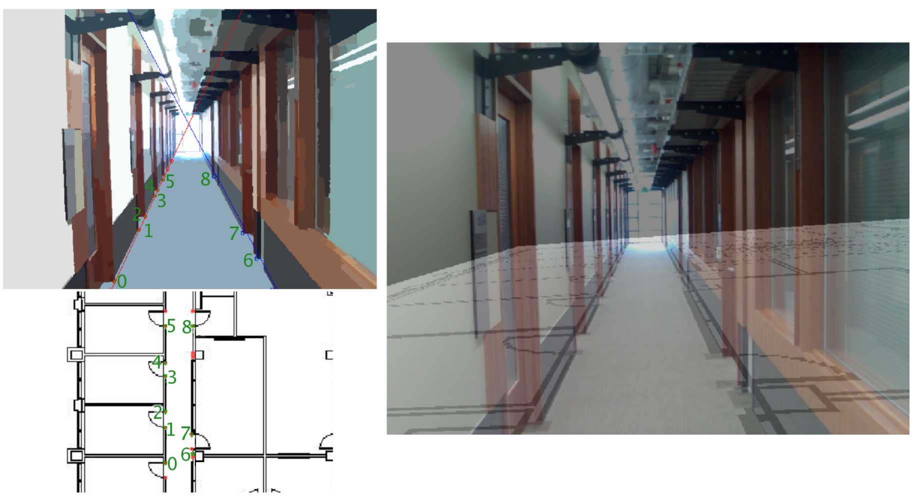

Fig. 6. Results of point correspondence algorithm. Upper left: the detected points on the image labeled by number. Lower left: the points on the floorplan that were matched to the image, labeled with corresponding numbers. Right: floorplan warped into the image space and overlaid on top of the original image. This example has 11 images points and 31 map points and completes matching in about 12 seconds. The view the opposite direction down the hallway is visually similar, so this is not a trivial example.

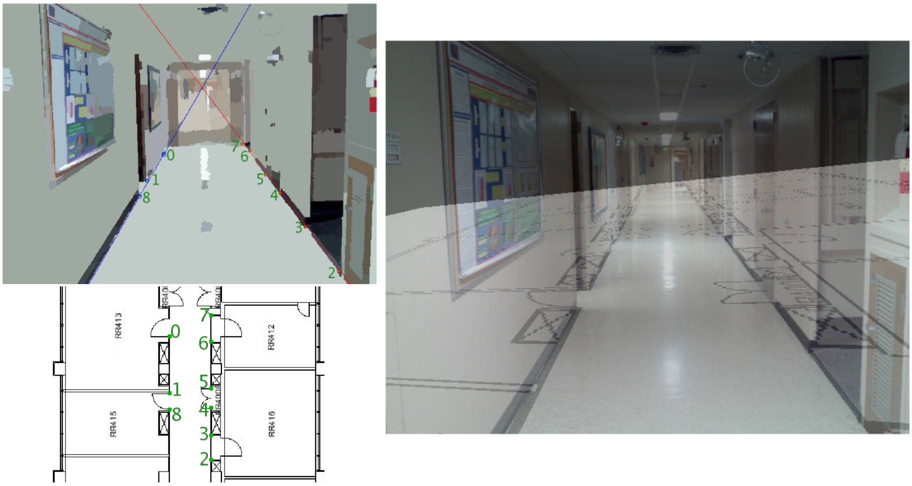

Fig. 7. Results of point correspondence algorithm. Upper left: the detected points on the image labeled by number. The detected points of this example required some manual filtering to get a good match. Lower left: the points on the floorplan that were matched to the image, labeled with corresponding numbers. Right: floorplan warped into the image space and overlaid on top of the original image. This example has 9 image points and 19 map points and completes matching in about 8 seconds.

image) are higher because they are likely to be detected more accurately in the image. The highest ranking solution at the end of the search is then used. See Figure 5 and 6 for results from our Computer Science building. Figure 7 shows results from our Health Sciences Center. This image presented more challenges due to the lower contrast, reduced brightness, and reflective floors. Although some manual filtering of the image features was required to get good a match to the floorplan, we believe it is possible to refine the algorithms to process these images correctly, as discussed in Section 4.

The RANSAC method returns results in 5 to 30 seconds on a 2.8GHz desktop machine, which is approaching the speed necessary for use in a live application. We believe that further improvements are possible that will increase both speed and robustness. This will be discussed more in Section 4.

3.2 Features in Heterogeneous Environments (Open Areas)

The system described above is tailored specifically to hallways and areas that have micro-landmark building features visible, but a lack of other distinct visual features. Images of rooms besides hallways often do not show many building features, but have a richness of other features, as in Figure 8. In order to support areas like this, a system like Photo Tourism can be used [4]. This requires a database of SIFT image features reconstructed from many images of the area and then aligned with the floorplan. New images can then be matched to this data and localized, providing a camera location and orientation. An example of this is shown in Figure 9, where the atrium of a building is analyzed using about 50 high quality (7 megapixel) images, and then a cell phone image is localized within this structure. The example shown here took about 7 seconds to match the cell phone image to the database of image features.

Approaches like Photo Tourism work well in open areas with many distinct image features, but have difficulty in more confined and uniform spaces like hallways. Additionally, it requires significant effort to take enough pictures to provide a database to localize new images. For this reason, we think a mixture of our hallway and floorplan micro-landmark system with an image feature texture-landmark database system will provide the most coverage at a lower startup cost. Integration of these systems may be done in a variety of ways, and is discussed further in Section 4.

Fig. 8. An atrium of a building. This lacks easily detectable building structure making it difficult to match directly to a floorplan. However, it has an abundance of distinct image features, making it possible to use other methods based on an image feature database.

3.3 Augmenting Images

The correspondence between the image features and the landmarks (from the floorplan or feature database) allow the camera pose can be estimated. This will give both the location and orientation of the camera in the space of the floorplan. The camera pose can be solved after finding the correspondence or the pose and correspondence can be solved simultaneously. It is also possible to use the simple homography calculated by our RANSAC matching method to map regions between the image and floorplan without further calculation.

The mapping between the image and floorplan can then be used to overlay information onto the camera image. Arrows to give navigation directions can be drawn on the floor by drawing the arrow on the floorplan and warping it into the image space. Tips of arrows can even disappear behind corners to give

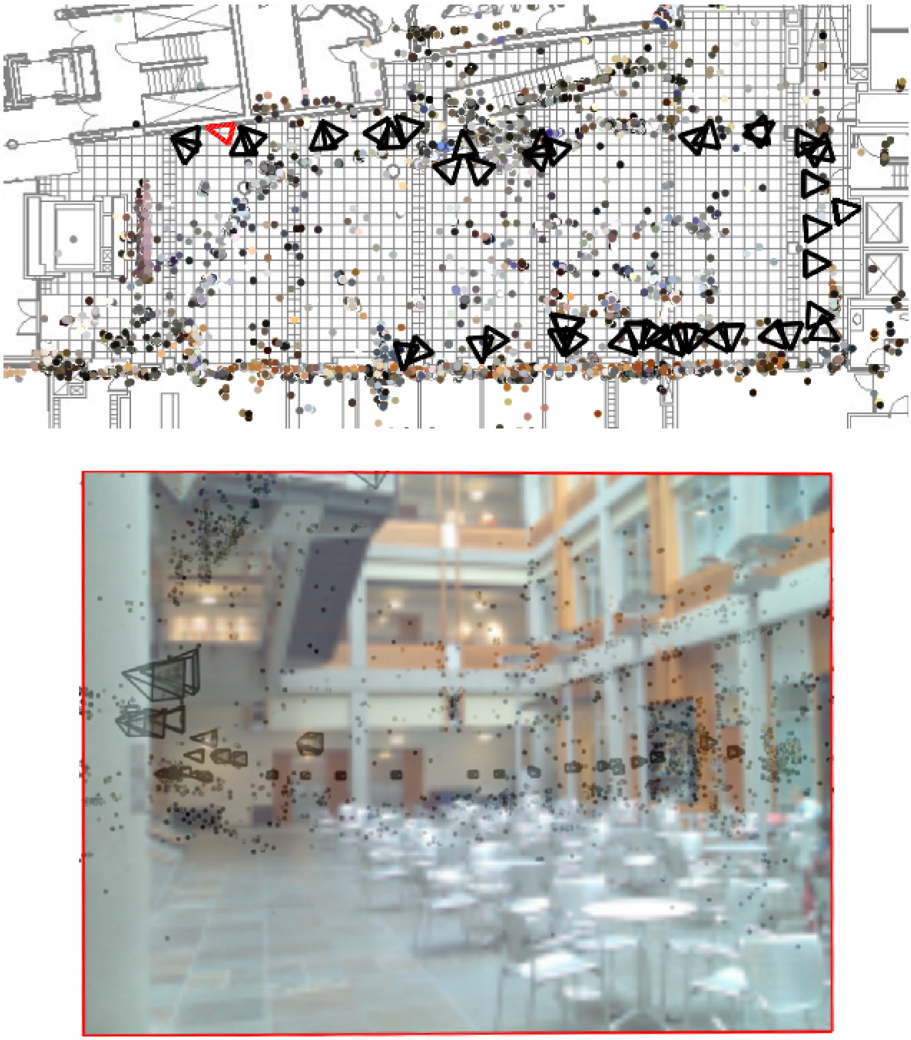

Fig. 9. This shows the reconstructed structure of a building atrium using Photo Tourism. The colored dots show the location of detected image features. The black triangles show the camera positions for images used in the reconstruction. The red triangle shows the location of the cell phone image within the reconstruction. Above: the reconstruction has been manually aligned with the building floorplan. Below: a view of the image is shown overlaid on the reconstruction.

an added sense of depth by clipping to the area segmented as floor, without requiring additional knowledge of the 3D structure. An example of this is shown in Figure 10. Since the location of doors is known in the image, the doors can be marked with additional labels, as suggested in Figure 1.

Fig. 10. This shows an arrow overlaid to match the perspective of the floor. The calculated homography was used to place the arrow.

4 Remaining Issues

We have demonstrated the feasibility of an image based localization system for overlaying information. However, many improvements are necessary in order to build a live application. Different areas of future work will be discussed in the following sections.

4.1 Robustness

This would not be an interesting problem if the simple methods proposed here worked for all situations. Indeed, there are many cases it does not work for that must be overcome. Figure 11 shows some possible problem areas. Transitions in floor covering can easily cause problems in segmenting the floor. Additionally, reflective floors make it difficult to segment the floor and produce reflected copies of relevant features. Combining information from probable floor edges as well as color and texture may help both problems, as could including information about floor types in the floorplan. Additionally, providing a small number of training examples may enable a customized doorway-corner feature detector while still maintaining low startup cost. Buildings with different types of doors or flooring in different areas could use a marked floorplan to select the appropriate detectors based on the rough location estimate.

Breaking assumptions about floor shape, for example hallways that curve, may cause problems with fitting lines to these edges, so the algorithms should not rely on this. Areas with few features, such as stretches of hallway without doors, or areas with unusual features, such as a catwalk, must be dealt with. Including floor edge lines as features in addition to points may provide more opportunity for matching, as would training specific doorway or corner detectors for each building. Estimates of user motion may also provide additional information about current location. These estimates may come from the location system, from the camera in a video mode, or even from external sensors, such as accelerometers and gyroscopes. Additionally, images that contain clutter will cause problems with simple segmentation. A box sitting on the floor may cause spurious features, as could a person walking by. In addition to improving the image analysis component, the point correspondence component should be robust to both spurious and missing features. Making more direct use of a camera model when matching between the image and the map should result in better constrained and more robust solutions. We are investigating the requirements of the matching system in order to inform the development of the feature detectors.

To cover a wider range of environments, a mixture of floorplan-based features and image based features can be used. Although navigation instructions are not as useful in a room, there are applications that could use this feature. There are few features in the floorplan that are visible in an average room, however the presence of objects in these rooms provides features that are more likely distinguishable with SIFT [5] and Photo Tourism [4]. In order for a system that uses SIFT image features to be robust to matching a wide variety of images, a large volume of pictures are necessary. In addition to variance in lighting at different times of day, camera quality also influences matching. For this reason, it may be necessary to include pictures of the environment across a range of lighting and a range of cameras. It may also be possible to use the floorplan feature system to bootstrap the SIFT feature collection, giving a lower startup cost but the ability to leverage SIFT features. Outdoor environments could also make use of SIFT features, but would not be able to leverage map information, instead requiring a large volume of images of the environment.

Even with robust algorithms, there are likely some images that will cause failure. It is important to be able to recognize these failures automatically and possibly provide a measure of uncertainty. Relaying this back to the user with information about what went wrong will increase their trust of the system by not giving obviously wrong results. It will also allow them to take this information into account when taking a new picture.

4.2 Speed

It is also important to consider response time in order to use this approach in an application. A user will likely not want to wait more than a few seconds for a response, so our goal is a response time of 5 seconds or less. If all the processing is done on a server and not on the phone, the time to send the image across a network must also be included. We assume a Wi-Fi enabled phone for the location system, and use 640x480 images, but other scenarios may take significantly more network time. Even for our case, algorithms must be chosen to meet this time goal. Our current bottleneck is finding correspondence between the features in the image and the map, but we believe this can be improved further. New methods for image analysis will also need to be chosen with speed in mind. It is also possible that some or all of the processing could be done on the phone as the processing power of phones continues to increase.

Fig. 11. Examples of images that pose problems for simple segmentation. Changes in floor material, reflective floors, and unusual structures such as curved walls and catwalks break simple assumptions, but they should all be handled properly. Additionally, people or clutter in the hallways will cause problems detecting features, as will environments with low contrast difference between floors and walls.

The Photo Tourism system is not designed to provide quick camera pose estimation within a large space given a new image and location estimate. It will be necessary to design the feature database to return a small number of features in the area around the location estimate in order to support our live user interaction model. Additional optimizations may be necessary to increase speed and improve handling of lower quality cell phone images. Additional work will be necessary to integrate both the floorplan-based landmarks and the image-based landmarks into one cooperative system. A simple geographic based approach could be used to dictate using one approach in hallways and the other in open areas, or the two types of features could be detected in parallel and then used together to solve for camera pose.

Video based "magic lens" or augmented reality type applications are also an area requiring optimization. Even if the initial camera pose takes some time to compute, assuming relatively small motion between frames it should be possible to compute an update to camera pose in a fraction of the time required for the general case. Since this system uses data in the image itself to calculate where to show things on the image, it should not be subject to the disconcerting lag in systems that use separate sensors to determine where the camera is aimed. An example of such a system is Nokia's MARA, which relies on GPS and compass sensors [19].

4.3 Applications

A system like this would make developing and using augmented reality applications for indoor environments much easier. Environments would not need to be instrumented with additional features; at a minimum a floorplan is required in order to deploy the system, although additional information such as photographs or property tags may be provided to improve robustness. Users would only need to have a camera phone to take advantage of the applications. Navigation applications could provide much clearer directions by overlaying directions on images of the current environment. Navigation applications could also make use of the improved location accuracy, again without the need for additional markers as proposed in other navigation systems [20]. Using images of the environment instead of words or maps is more effective for people with cognitive disabilities [1], and would be a pleasant interface for most users. The ubiquity of cellphone usage would not make users requiring assitance stand out. Pausing to take a picture of the current location also seems like a natural way to ask for help in deciding where to go next. This interface may also be a convenient way to retrieve timely information about the current environment. Linking information from a calendar system would enable it to show the availability of the nearby conference room, without requiring data entry on the phone. An X-Ray Vision-type application could use more detailed information about the building to show plumbing or similar structures within the walls. Overlaying those details on images of the walls will allow servicemen to more accurately predict where work might be needed.

Outdoor environments would also benefit from navigation instructions tailored to the current view, and we believe a similar approach will work in outdoor environments. It could also be used as a way to identify landmarks and index into information about them. For example, it could find the name or address of a building, hours of operation, reviews from other people about that location, and other information–all without requiring users to do complex lookups on their phones. There is a wealth of opportunities for building applications on top of a system like this.

5 Conclusion

We have presented early results that demonstrate it is possible to provide the capability of information overlay on camera phones for a variety of indoor environments. Our system tailored to hallway environments currently provides reasonable latency, and still has room for further improvement. We have also demonstrated that a system based on SIFT image features can work in more open environments, and that these two systems complement each other. Although there are significant improvements necessary to achieve a workable system, we have developed a framework to support localization from images and allow applications such as navigation and dynamic directory assistance to overlay information on a user's camera phone with minimal infrastructure cost.

Acknowledgements

This work is supported by NIDRR under our ACCESS project. Thanks to Noah Snavely, Steve Seitz, Linda Shapiro, Alan Liu, and Dan Goldman for their assitance on this project.

References

1. Liu, A.L., Hile, H., Kautz, H., Borriello, G., Brown, P.A., Harniss, M., Johnson, K.: Indoor wayfinding: developing a functional interface for individuals with cognitive impairments. In: Assets '06: Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility, pp. 95–102. ACM Press, New York, USA (2006)
2. Ekahau Wi-Fi positioning engine, a commercial location product, `http://www.ekahau.com/`
3. Ferris, B., Haehnel, D., Fox, D.: Gaussian processes for signal strength-based location estimation. In: Proceedings of Robotics: Science and Systems (August 2006), Philadelphia, USA (2006)
4. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3d. ACM Trans. Graph 25(3), 835–846 (2006)
5. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision 60(2), 91–110 (2004)
6. Kosaka, A., Pan, J.: Purdue experiments in model-based vision for hallway navigation. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 1, 87–96 (1995)
7. Karlsson, N., di Bernardo, E., Ostrowski, J., Goncalves, L., Pirjanian, P., Munich, M.: The vslam algorithm for robust localization and mapping. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation (April 2005), pp. 24–29 (2005)
8. Dellaert, F., Burgard, W., Fox, D., Thrun, S.: Using the condensation algorithm for robust, vision-based mobile robot localization. In: Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (1999)
9. Wolf, J., Burgard, W., Burkhardt, H.: Robust vision-based localization for mobile robots using an image retrieval system based on invariant features. In: Proceedings of the IEEE International Conference on Robotics and Automation (2002)

10. Fritz, G., Seifert, C., Paletta, L.: A mobile vision system for urban detection with informative local descriptors. In: ICVS '06: Proceedings of the Fourth IEEE Intl Conference on Computer Vision Systems, IEEE Computer Society, Los Alamitos (2006)
11. Rekimoto, J., Ayatsuka, Y.: Cybercode: Designing augmented reality environments with visual tags. In: Proceedings of DARE (2000)
12. Hile, H., Kim, J., Borriello, G.: Microbiology tray and pipette tracking as a proactive tangible user interface. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 323–339. Springer, Heidelberg (2004)
13. Caudell, T., Mizell, D.: Augmented reality: an application of heads-up display technology tomanual manufacturing processes. Proceedings of Hawaii International Conference on System Sciences. 2, 659–669 (1992)
14. Köhler, M., Patel, S., Summet, J., Stuntebeck, E., Abowd, G.: Tracksense: Infrastructure free precise indoor positioning using projected patterns. In: Pervasive Computing (2007)
15. Hecht, B., Rohs, M., Schöning, J., Krüger, A.: Wikeye - using magic lenses to explore georeferenced wikipedia content. In: 3rd International Workshop on Pervasive Mobile Interaction Devices (PERMID 2007) (2007)
16. Comaniciu, D., Meer, P.: Mean shift analysis and applications. In: ICCV '99: Proceedings of the International Conference on Computer Vision, vol. 2, IEEE Computer Society, Los Alamitos (1999)
17. Guru, D., Dinesh, R., Nagabhushan, P.: Boundary based corner detection and localization using new 'cornerity' index: A robust approach. In: CRV '04: Proceedings of the 1st Canadian Conference on Computer and Robot Vision (CRV'04), pp. 417–423. IEEE Computer Society, Los Alamitos (2004)
18. Rous, M., Lupschen, H., Kraiss, K.: Vision-based indoor scene analysis for natural landmark detection. In: IEEE International Conference on Robotics and Automation, pp. 4642–4647 (April 2005)
19. Nokia mobile augmented reality applications (MARA), `http://research.nokia.com/research/projects/mara/index.html`
20. Muller, H.J., Schoning, J., Kruger, A.: Mobile map interaction - evaluation in an indoor scenario. In: Workshop,Mobile and Embedded Interactive Systems (MEIS'06), part of Informatik (October 2006)