Hongwen Kang, Alexei A. Efros, Martial Hebert, Takeo Kanade, “Image Matching in Large Scale Indoor Environment,” *The IEEE International Conference on Computer Vision and Pattern Recognition*, June, 2009.

Image Matching in Large Scale Indoor Environment

Hongwen Kang Alexei A. Efros Martial Hebert Takeo Kanade
School of Computer Science
Carnegie Mellon University
{hongwenk, efros, hebert, tk}@cs.cmu.edu

Abstract

In this paper, we propose a data driven approach to first-person vision. *We propose a novel image matching algorithm, named* Re-Search*, that is designed to cope with self-repetitive structures and confusing patterns in the indoor environment. This algorithm uses state-of-art image search techniques, and it matches a query image with a two-pass strategy. In the first pass, a conventional image search algorithm is used to search for a small number of images that are most similar to the query image. In the second pass, the retrieval results from the first step are used to discover features that are more distinctive in the local context. We demonstrate and evaluate the* Re-Search *algorithm in the context of indoor localization, with the illustration of potential applications in object pop-out and data-driven zoom-in.*

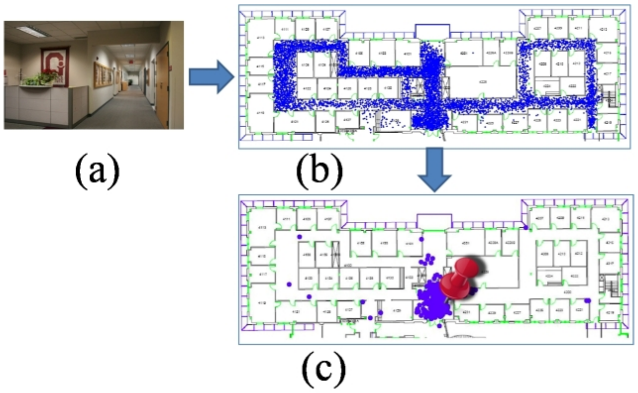


Figure 1. Example application: image search for indoor localization. (a) the input image, capturing environment seen by a user; (b) a large number of images pre-captured in the same environment, with each dot corresponds to a picture; (c) the location of the user is determined by finding images that are matched with the input image.

1. Introduction

Unlike traditional vision systems based on fixed cameras that observe people from the outside, first-person vision systems, in which a camera observes the environment from each user’s point of view, are able to work with data that relates directly to the user’s interests and intentions.

In this paper, we focus on one mode of usage of first-person systems in which information is extracted by comparing the image viewed by the system with a large collection of pre-recorded images. A standard example is a *localization* scenario in which the most similar images to the currently observed image are retrieved from the pre-recorded collection and the associated recorded positions are reported to the user.

Fig. 1 shows an example: given an input image of the environment seen by the user (Fig. 1(a)), we can match it to a large database of pre-captured images that are annotated with locations (Fig. 1(b)) and find these ones that capture the same scene and predict where the user is (Fig. 1(c)).

The key building block in this, and other similar applications, is image matching, *i.e*., the ability to define a similarity metric between images and to retrieve the most similar images from a large collection based on this metric. Although there is a substantial amount of prior work in image retrieval and search, attempting to find same buildings [20] or objects [19], it tends to focus on outdoor environments with distinctive structure and unique landmarks.

Instead, in this paper, we explore the image matching problem in the context of typical indoor environments. Unlike outdoor environments, manmade indoor environments are usually full of self-repetitive structures and confusing patterns (Fig. 2), which make them extremely challenging for image matching. In Fig. 2, for example, many of the images look similar at first but, upon closer inspection, one can notice subtle details that distinguish each image from the others.

By analogy with this discovery process, in this paper, we approach this image matching problem by following a two-pass strategy, named ***Re-Search***. In the first pass we would like to efficiently retrieve a number of images that are most similar to a query image, one can use state-of-art

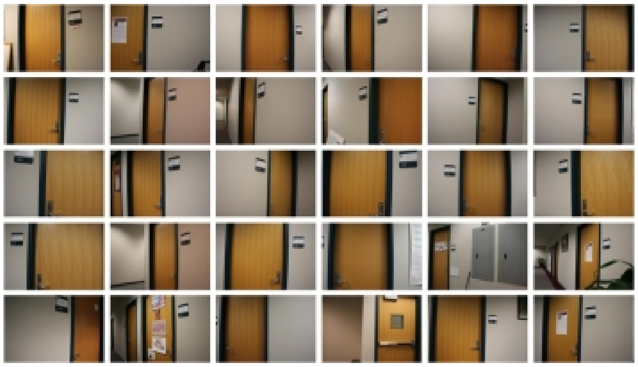


Figure 2. Indoor environment is full of self-repetitive structure and confusing patterns, a particular case is that all doors look similar.

image search algorithms to accomplish this. The rationale is that, for a large database, this step can very efficiently filter out a large number of images that are obviously unrelated to the query image. And in return, it gives a small number (*e.g.* 50) of candidates that appear to be the most similar to and therefore easily confused with the query image. We use these candidates as a context to help us discover more subtle details of the images. These details were overlooked in the first pass, because on a global scale they are not as distinctive. For example in Fig. 2, all the features from the door handles and nameplates are very distinctive in distinguishing door image from non-door images, while some other features, such as the posters on the doors are the ones that eventually distinguish one door from the others.

There are many ways one can implement this *Re-Search* approach, in particular, in Section 2 we discuss our implementation following the *TF-IDF* scheme originated from textual retrieval community [1, 14] and was introduced to image search in [24]. Section 3 reviews related works in the image search domain. We show an example application of image search for localization in Section 4, which also demonstrate the effectiveness of our *Re-Search* approach. We conclude and give further discussion of our approach and future works in Section 5.

2. Re-Search

In our approach, we adopt a Bag-of-Words model. We assume that standard visual features are extracted, and defer the details to Section 4, where we discuss specific application. First, a set of visual words is learned by clustering features extracted from the database, with each cluster center corresponding to a visual word. Each feature is assigned a visual word ID, corresponding to the closest cluster center under Ecliduean distance. An image can therefore be represented by a set of visual words. This step is normally referred to as *vector quantization*.

There are various ways to measure the similarity of two images, given the discrete visual words representation. One straightforward way is to measure the distance of normalized visual word histograms, which gives more weight to words that appear more frequently in that document. However, like in text retrieval, where some words are seen in most documents, *e.g.* “the”, “a”, *etc*., this is also commonly observed in visual words representation [24]. Therefore, it is more advantageous to down-weight less distinctive words that appear frequently in most images, and up-weight words that only appear in very few images. This combined scheme is referred to as *Term Frequency-Inverse Document Frequency (TF-IDF)* [1, 14, 24]. More formally, given a word (term) *t* in document *d*, its TF-IDF weight is given by:

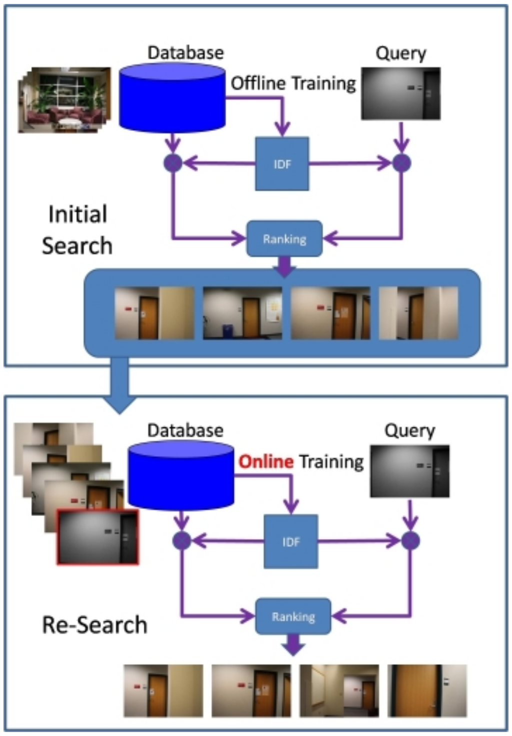


Figure 3. *Re-Search* process diagram. The first pass makes use of state-of-art image search algorithms to find a small number of images from the whole database, which are the most similar to the query image [19]. In a second pass, a new *idf* term is learned using (3) from this set of images.

$$tf\text{-}idf_{t,d} = tf_{t,d} \times idf_t. \tag{1}$$

$tf_{t,d}$ is defined as:

$$tf_{t,d} = \frac{n_{t,d}}{\max_l n_{l,d}}, \tag{2}$$

where $n_{t,d}$ and $n_{l,d}$ are the number of occurrences of the words t and l in document d, respectively.

idf_t is calculated based on a global statistics over all documents:

$$idf_t = \frac{\sum_{l,d} n_{l,d}}{\sum_d n_{t,d}}. \tag{3}$$

Under this definition, we now treat images and their TF-IDF vector representations equivalently. The similarity between a query image $\vec{q}$ and an image $\vec{d_j}$ in the database can be calculated using the *cosine similarity* measure [1]:

$$sim(\vec{q}, \vec{d_j}) = \frac{\vec{q} \cdot \vec{d_j}}{|\vec{q}| \times |\vec{d_j}|}. \tag{4}$$

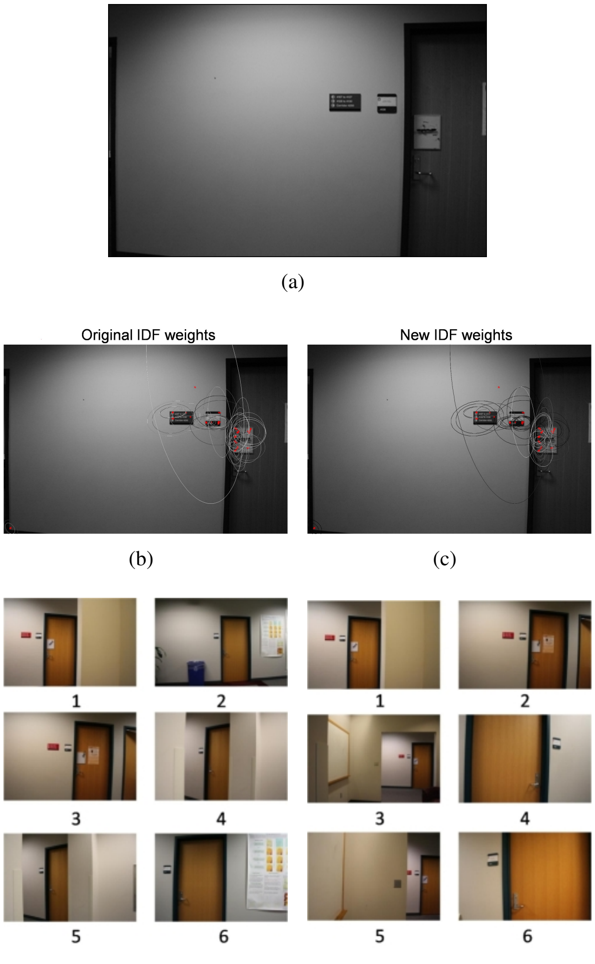
However, this similarity measure retrieves only approximately similar images. It is essentially because this similarity measure uses statistical information that only reflects global properties of the database. More specifically, the IDF measure in (3) essentially measures scarceness of a feature with respect to the whole database. However, features that are statistically rare with respect to the whole database, can be overly abundant when looking at a smaller subset (Fig. 2). In order to distinguish these very similar instances, one has to depend on more subtle details in the local neighborhood context.

In this paper, we propose a two-pass approach, named *Re-Search*, that combines both global and local visual word statistics. As shown in Fig. 3, the first pass makes use of state-of-art image search algorithms to find a small number of images from the whole database, which are the most similar to the query image. This set of images is used in the second pass as the new training database. A new idf term is learned using (3) from this small set of images, instead of from the whole database. This way, we gather statistics of visual words distribution in the local neighborhood and use that to discover features that are more distinctive among these similar images.

Fig. 4 shows an example of how effective this approach is in finding exactly matched objects in this highly confusing environment. Comparing the weights of features in this example, in the first round of image retrieval, those features that are effective in distinguishing doors from other objects are given high weights in idf vector. While in the *Re-Search* step, we have all door related images retrieved, and therefore higher weights are given to other features that are rarer locally and therefore more effective in distinguishing one door from the other doors.

3. Survey of related image search techniques

Recent years have seen huge progress in image search. While the dataset size has been increased from a few thousand [24] to millions [7, 9, 19, 21], the use of advanced data structures and indexing algorithms reduces the computational complexity by up to 1000 times, making it fast enough for interactive use [4, 19]. In this section we survey recent advance in image search algorithms that are closely related to our work.



(a)

(b)

(c)

(d)

(e)

Figure 4. An example of the IDF weights and the retrieval results in the initial search and the *Re-Search* steps. (a) A query image; (b) The original IDF weighting used in the initial search step, the brighter the ellipses are, the higher the weights are; (c) the new IDF weighting used in our *Re-Search* step, the brighter the ellipses are, the higher the weights are; (d) the initial retrieval and ranking; (e) the new retrieval and ranking after the *Re-Search* step.

3.1. Feature extraction

In the image matching task, given a query image, we want to find images captured at the same location (exact matching) or similar environment (similarity matching). Since the viewpoints and scales of the database images and the input query images are often different, it is not likely that any pixel-wise similarity functions can give us reasonable similarity measures. Instead, it is possible to find parts of an image that are both distinctive and robust across different view points, scales. Image information inside these regions is a good approximation of the overall appearance of the objects and scenes [16, 17, 18]. Measuring the visual similarity of these regions usually approximates how similar these images are.

Many approaches have been devoted to detecting this type of regions, an incomplete list of these region detectors and their representative applications includes *Difference of Gaussians(DoG)* [13], *Harris-Affine(HARAFF)* regions [6, 16, 24, 25, 26], *Hessian-Affine(HESAFF)* regions [3, 9, 10, 16, 20, 21, 22], *Maximal Stable Extremal Regions(MSER)* [15, 19, 24, 25, 26, 29]. For more details, the readers are referred to an extensive comparison of region detectors conducted in [18].

Regardless of the specific region been used, one can represent each region using Scale Invariant Feature Transformation(SIFT) [13], which transforms an image region into a 128 dimensional feature vector.

3.2. Image search algorithms

For a relatively small database, which consists of thousands of images, a few thousand visual words will probably be enough [24]. However when the size of database increases, this becomes soon inadequate. First, the more images are stored in the database the more variance there is in the visual appearance. As a result, more visual words are needed to faithfully represent this dynamic range of visual variance. Consequently, this becomes intractably high dimensional for vector quantization (*e.g.*, comparing a feature vector with millions of visual word clusters).

Various sophisticated data structures have been introduced. Recently multi-level vocabulary trees [19] have been proposed for image retrieval in large database. Compared to linear scan, it has $log(n)$ computational complexity in vector quantization. The ability to handle large number of visual words then boosts image matching quality.

Along this line, it was shown in [20] that using the Approximate K-Means (AKM) algorithm instead of the Hierarchical K-Means (HKM) algorithm [19] enabled the use of a large number of visual words and made the vector quantization more robust. This quantization issue was further addressed in [21], where a soft assignment strategy was proposed to map each visual feature to multiple visual words instead of committing to just one. A novel Hamming code embedding strategy and a combination of coarse/fine feature distances were proposed in [9] as a way to compensate quantization noise.

It was discovered in [10] that by iteratively adjusting the asymmetrical neighborhood structure one could enhance neighborhood symmetry property and therefore could generate a distance measure that was more suitable for similar image search. The min-Hash function was proposed in [4], it provided a way to efficiently search through large scale database, and required only a small amount of data to be stored.

However, none of the work to our knowledge has addressed the issue of combining both global and local statistics in finding exact match of images. A similar line of research follows metric optimization and per-example based local distance function learning, such as [5]. However, their approaches were primarily designed for classification tasks.

In the text retrieval domain, our work is closely related to the relevance feedback and query expansion approaches [2, 8, 11, 23, 30]. In these approaches, it was assumed that one can determine the relevance between query and retrieval results, either as explicitly as from a user's input, or as implicitly as defining relevant retrieval to be the top ranked results. This relevance feedback information was used to generate a new query that fitted better with the searcher's intention. While in our case, we do not assume any such feedback, except that we assume that the dataset is composed of confusing image samples and that the goal is to find exact matches.

4. Example application: image search for indoor localization

Here we show how the image matching algorithm is evaluated in the localization scenario. We use a database of images that have been pre-annotated with locations. Given a single image, our image matching algorithm searches through the database and finds the images in the database that capture the same scene. Using the pre-annotated location information, we can estimate the location where the input image was taken.

For feature extraction, we used the Harris-Affine (HARAFF) region detector [16], combined with SIFT descriptor [13]. On average, we detected about 500 to 1000 HARAFF regions per image, which had a resolution of 640×427 pixels. For efficient indexing and searching, we learned a vocabulary tree from the database, with a fan-out factor of 4 and depth of 10. In the initial search step, our weighting scheme follows the TF-IDF definition. Though information from multiple levels of the tree could be summarized together to get a more balanced measure, in our implementation, we only use the weights from the leaf level [19]. In the *Re-Search* step, we learn new idf weights based on the top 50 retrievals.

We also find that there are several ways to improve the robustness of the *Re-Search* algorithm. First, using $log\ idf$ space is helpful in improving robustness [19, 28]. Also it is useful to compute the new idf term as a linear combination of the idf_{local}, estimated from the local neighborhood and the idf_{global} which is estimated from the whole database and have been used in the initial search, *i.e.*,

$$idf' = \alpha \cdot idf_{global} + (1 - \alpha) \cdot idf_{local}, \qquad (5)$$

where idf' is the new idf weight used in *Re-Search*, we set $\alpha = 0.05$ in all the experiments. This is useful to prevent a degenerate case of the *Re-Search* algorithm, *i.e.* when most of the retrieved images are captured from the same location, noise could be amplified by the idf measure and, as a result, the irrelevant images could be selected as the top ranked

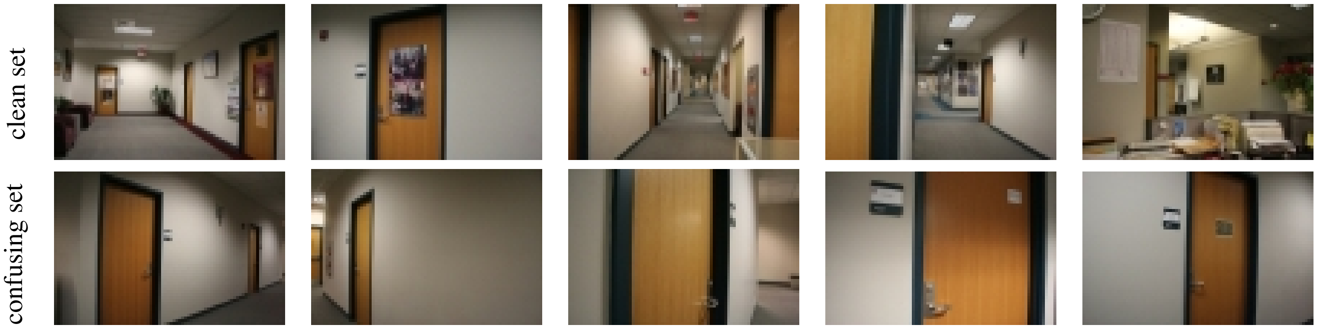
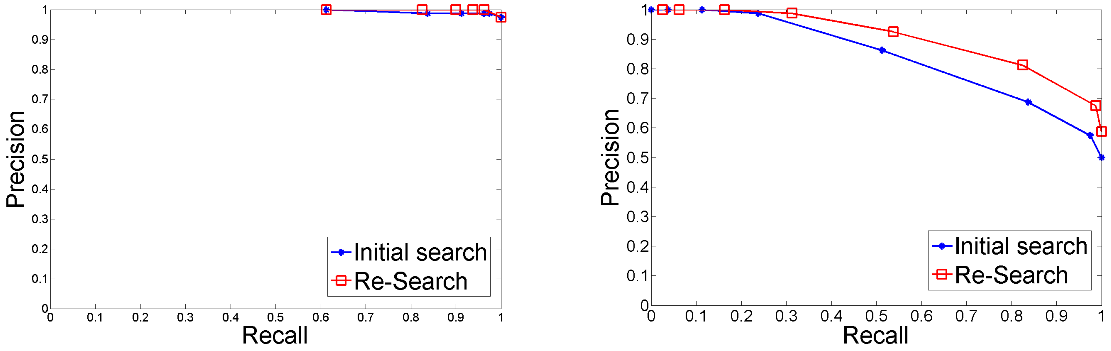


Figure 5. Some examples of the "clean set" (images with rich and distinctive visual structures) and the "confusing set" (images that capture more detailed part of the scene, or scenes with objects that could easily be found somewhere else, such as doors).



(a)

(b)

Figure 6. Quantitative analysis of the localization performance after the initial image search step and the *Re-Search* step. (a) the Precision-Recall curve on the "clean set". (b) the Precision-Recall curve on the "confusing set".

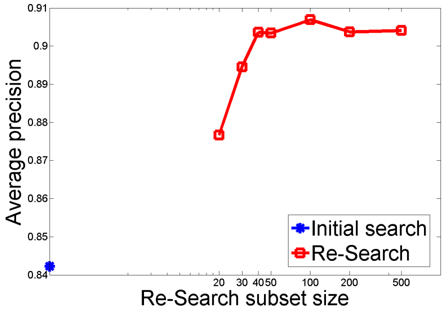
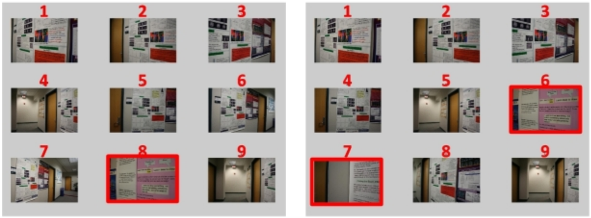


Figure 7. The effect of changing the subset size on the *Re-Search* performance. For clarity, the scale is enlarged compared to Fig. 6.

results (Fig. 8). Another way to handle this is damping the *idf* calculation by removing visual words that have less than a minimum number of occurrences [11].

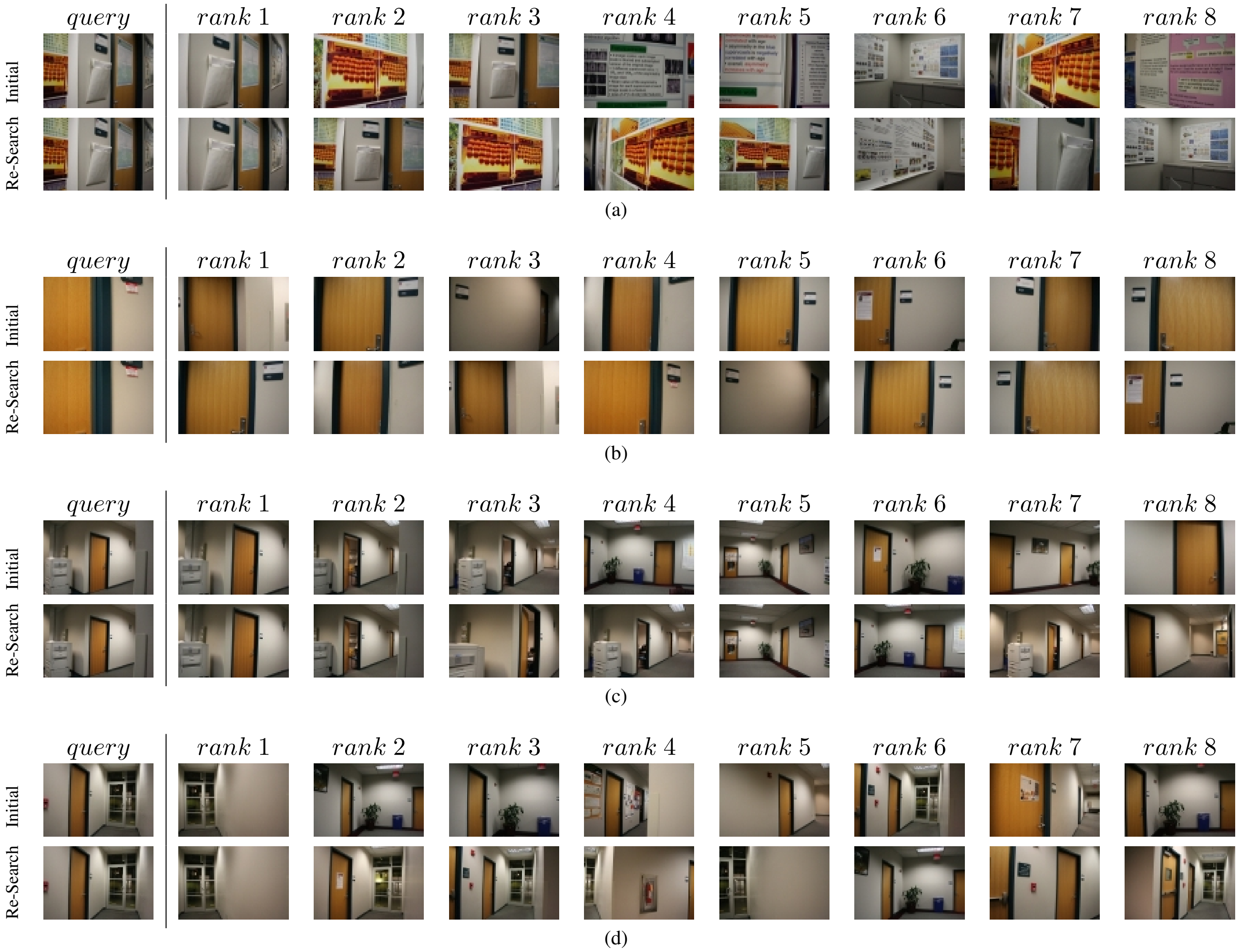
We evaluate performance of the localization algorithm under a precision-recall formulation. For an input image, 8 of the most similar pre-captured images are retrieved. A potential localization is suggested if there are a cluster of pre-recorded images, denoted $\mathcal{R}$, captured less than 3 meters away from each other among the retrieved images. If there are more than one cluster, the larger and higher ranked



(a) Initial search result

(b) *Re-Search* result

Figure 8. An example illustrating the robustness of the *Re-Search* algorithm. In both (a) and (b), image 1 is the query. In the initial result (a), most of the retrieved images capture the same scene as the query, except image 8. In (b), the *Re-Search* result is worse, without using the prior information as shown in equation (5), *i.e.* more irrelevant images (image 6 and 7) are retrieved and the outlier (image 8 in (a)) is now given a higher rank, *i.e.* from 8 to 6. [Best viewed in color.]



(a)

(b)

(c)

(d)

Figure 9. Some qualitative analysis of the image matching results after the initial image search step and the proposed *Re-Search* step.

set is picked. The size of set $\mathcal{R}$ is denoted $|\mathcal{R}|$. By adjusting a threshold, denoted $|\mathcal{R}|_T$, on the minimum size of $\mathcal{R}$, one can change the confidence of each localization prediction. Higher $|\mathcal{R}|_T$ gives more confident prediction, and results in higher precision. On the contrary, lower $|\mathcal{R}|_T$ results in less confident decision, but it gives higher recall. If the threshold $|\mathcal{R}|_T == 1$, then the location of the top ranked image is used for prediction. If no set satisfies this minimum requirement, then localization fails. A false negative is detected for any query image that has a matching location in the database. For cases that satisfy the minimum requirement, false positives are detected as those where the minimum location distances between the query image and the images in $\mathcal{R}$ are more than 3 meters.

Our database consists of around 8.8 thousand images (each associated with a location label). We built an automated data collection rig to capture these images, users who were familiar with this environment helped to build the ground truth image-location correspondences (Fig. 1(b)).

For the testing set, we captured one set of images with rich and distinctive visual structures, we call this set the "clean set". It acts as a control set, and measures how our system performs under normal condition. Also, we captured a much more challenging set of images that captured more detailed part of the scene, or scenes with objects that could easily be found somewhere else, such as doors. We call this set the "confusing set". This set represents the scenarios when the user is close up to some specific objects in the scene. Both of these sets are composed of 80 images, examples of them are shown in Fig. 5. We exclude cases that are not possible to distinguish without higher level knowledge, such as doors with only nameplates, which are non-distinguishable without character recognition.

Fig. 6(a) and 6(b) show the Precision-Recall curves of initial search and after using our *Re-Search* method on both of the testing sets. On the "clean set", both the initial search and *Re-Search* achieve almost perfect performance, with close to 1% gain using our approach. This demonstrates the practical usefulness of our system in indoor localization.

What is of more interest is to test the robustness of the system in handling the "confusing" situation, because this is the scenario in which a user will rely on the system the most. We can see that compared to the performance in the "clean set", initial search performance degrades severely, while after using *Re-Search* algorithm we still achieved about 85% precision at 80% recall, with a maximum of about 15% gain in precision compared to initial search result.

The size of the subset used in the *Re-Search* step is also an important factor. Too small a subset may exclude many actual relevant images. It is not a good idea to use an overly large subset either. The extreme case would be using the whole database for *Re-Search*, in which case no improvement can be made because the idf will be unchanged. Fig. 7 shows the change of performance through varying the *Re-Search* subset size. It suggests that the performance reaches a relatively stable level as soon as we use more than 50 top ranked results from the initial search. In our experiment, we found that a subset with size 50 seems to be a good balance of computational performance ($3 \sim 4$ seconds using unoptimized Matlab implementation) and matching quality.

Fig. 9 shows more qualitative comparisons. Our approach retrieves more relevant images and also gives better ranking results.

5. Discussions and conclusions

In this paper, we developed a novel way of capturing the visual information in an indoor environment, from the first-person's viewpoint. By taking a large number of images, it covers almost every corner of the environment of people's daily life.

We proposed a novel image matching algorithm, named *Re-Search*, which emphasizes robustness for confusing self-repetitive patterns of indoor environment. We demonstrated its effectiveness through an indoor localization application.

There are some limitations that we would like to address in the future. First, in our image matching algorithm we are using the standard salient feature detector (HARAFF). It is effective in detecting features that are salient and repetitive across views. However, in the indoor environment, many parts of the environment are textureless. This could be very useful information, but is normally discarded. Some dense feature [12] or MSER type features would probably be helpful if combined with the corner type features.

It is interesting to compare our approach with other algorithms that use explicit geometrical epipolar constraints, in respect to space/time complexity, and reranking quality. We would like to optimize the *online Re-Search* algorithm further for realtime performance. Also, some of the parameters are selected and fixed empirically in our algorithm. It is interesting to learn statistical models that can adapt these parameters to the quality of the initial results automatically.

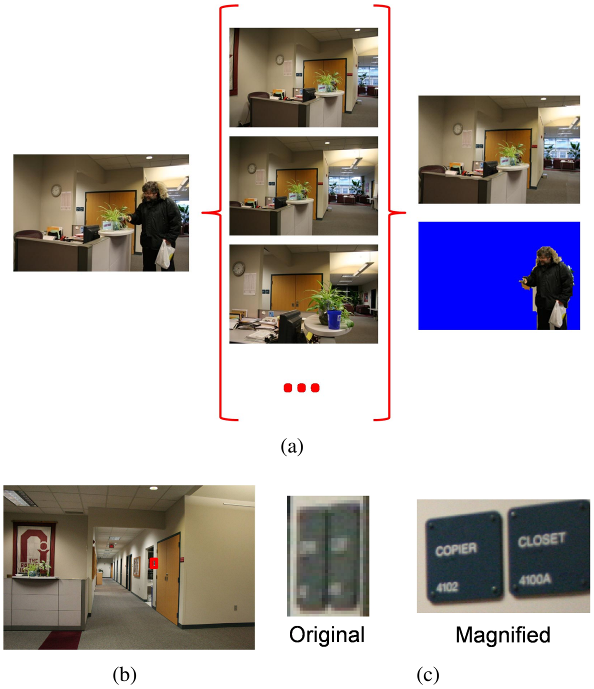


Figure 10. Potential applications that benefit from robust image matching. (a) background subtraction and object pop-out by matching the input image with large number of images captured beforehand. (b) the user selects part of the image (in red rectangle) that she wants to see more clearly; (c) our program generates a magnified view using the image that is matched to the input image, but with a much more closed-up view of the selected region.

For the localization application, we rely right now on users' label to generate correspondence between database images and points on the floor plan. It will be necessary to explore ways to generate this ground truth automatically, such as by using the structure from motion techniques to recover the relative locations of the cameras [27].

Beyond localization, reliable image matching techniques pave the way for numerous other applications. Fig. 10 demonstrates some potential applications and our work in progress, including background subtraction (Fig. 10(a)), and data-driven zoom-in view generation (Fig.10(b), 10(c)).

Acknowledgements

The authors would like to thank anonymous reviewers for helpful and constructive suggestions. Hongwen Kang is grateful to Christos Faloutsos for inspirational discussions. This work is based on work supported by the National Science Foundation under Grant No. EEEC-0540865.

References

[1] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, May 1999. 2, 3

[2] S. Chakrabarti, B. Dom, R. Agrawal, and P. Raghavan. Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies. *The VLDB Journal*, 7(3):163–178, 1998. 4

[3] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *Proceedings of the 11th International Conference on Computer Vision, Rio de Janeiro, Brazil*, 2007. 4

[4] O. Chum, J. Philbin, and A. Zisserman. Near duplicate image detection: min-hash and tf-idf weighting. In *Proceedings of the British Machine Vision Conference*, 2008. 3, 4

[5] A. Frome, Y. Singer, and J. Malik. Image retrieval and classification using local distance functions. In *Advances in Neural Information Processing Systems 19*, pages 417–424. MIT Press, Cambridge, MA, 2007. 4

[6] K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. *Computer Vision, IEEE International Conference on*, 2:1458–1465, 2005. 4

[7] J. H. Hays and A. A. Efros. Im2gps: estimating geographic information from a single image. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, June 2008. 3

[8] Y. Ishikawa, R. Subramanya, and C. Faloutsos. Mindreader: Querying databases through multiple examples. In *VLDB '98: Proceedings of the 24rd International Conference on Very Large Data Bases*, pages 218–227, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc. 4

[9] H. Jegou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *European Conference on Computer Vision*, LNCS. Springer, oct 2008. to appear. 3, 4

[10] H. Jegou, H. Harzallah, and C. Schmid. A contextual dissimilarity measure for accurate and efficient image search. In *IEEE Conference on Computer Vision & Pattern Recognition*, jun 2007. 4

[11] K. L. Kwok. A new method of weighting query terms for ad-hoc retrieval. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 187–195, New York, NY, USA, 1996. ACM. 4, 5

[12] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2169–2178, Washington, DC, USA, 2006. IEEE Computer Society. 7

[13] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004. 4

[14] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008. 2

[15] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, September 2004. 4

[16] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004. 3, 4

[17] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 27(10):1615–1630, 2005. 3

[18] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1/2):43–72, 2005. 3, 4

[19] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2161–2168, Washington, DC, USA, 2006. IEEE Computer Society. 1, 2, 3, 4

[20] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007. 1, 4

[21] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 3, 4

[22] J. Philbin and A. Zisserman. Object mining using a matching graph on very large image collections. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, 2008. 4

[23] I. Ruthven and M. Lalmas. A survey on the use of relevance feedback for information access systems. *Knowl. Eng. Rev.*, 18(2):95–145, 2003. 4

[24] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, volume 2, pages 1470–1477, Oct. 2003. 2, 3, 4

[25] J. Sivic and A. Zisserman. Video Google: Efficient visual search of videos. In *Toward Category-Level Object Recognition*, volume 4170 of *LNCS*, pages 127–144. Springer, 2006. 4

[26] J. Sivic and A. Zisserman. Efficient visual search for objects in videos. *Proceedings of the IEEE*, 96(4):548–566, 2008. 4

[27] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3d. In *SIGGRAPH Conference Proceedings*, pages 835–846, New York, NY, USA, 2006. ACM Press. 7

[28] R. O. Stehling, M. A. Nascimento, and A. X. Falc ao. A compact and efficient image retrieval approach based on border/interior pixel classification. In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 102–109, New York, NY, USA, 2002. ACM. 4

[29] A. Toshev, J. Shi, and K. Daniilidis. Image matching via saliency region correspondences. In *Computer Vision and Pattern Recognition, 2007. IEEE Conference on*, pages 1–8, June 2007. 4

[30] L. Wu, C. Faloutsos, K. P. Sycara, and T. R. Payne. Falcon: Feedback adaptive loop for content-based retrieval. In *VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases*, pages 297–306, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. 4