Accurate Localization by Fusing Images and GPS Signals

Kumar Vishal C. V. Jawahar Visesh Chari
Centre for Visual Information Technology,
IIIT Hyderabad, India

Abstract

Localization in 3D is an important problem with wide ranging applications from autonomous navigation in robotics to location specific services on mobile devices. GPS *sensors are a commercially viable option for localization, and are ubiquitous in their use, especially in portable devices. With the proliferation of mobile cameras however, maturing localization algorithms based on computer vision are emerging as a viable alternative. Although both vision and* GPS *based localization algorithms have many limitations and inaccuracies, there are some interesting complimentarities in their success/failure scenarios that justify an investigation into their joint utilization. Such investigations are further justified considering that many of the modern wearable and mobile computing devices come with sensors for both* GPS *and vision.*

In this work, we investigate approaches to reinforce GPS *localization with vision algorithms and vice versa. Specifically, we show how noisy* GPS *signals can be rectified by vision based localization of images captured in the vicinity. Alternatively, we also show how* GPS *readouts might be used to disambiguate images when they are visually similar looking but belong to different places. Finally, we empirically validate our solutions to show that fusing both these approaches can result in a more accurate and reliable localization of videos captured with a Contour action camera, over a 600 meter long path, over 10 different days.*

1. Introduction and Related Work

Localization refers to the idea of "locating" the position of an object within its environment. It has numerous applications in wearable computing, robotics, entertainment devices and consumer electronics. Most popular localization approaches are designed to represent object location in 3D coordinate systems either using the lat/long format like Global Position System (GPS) sensors, or by using metric distances like vision based localization methods. GPS based methods provide global/absolute information about the location of an object with the help of special purpose

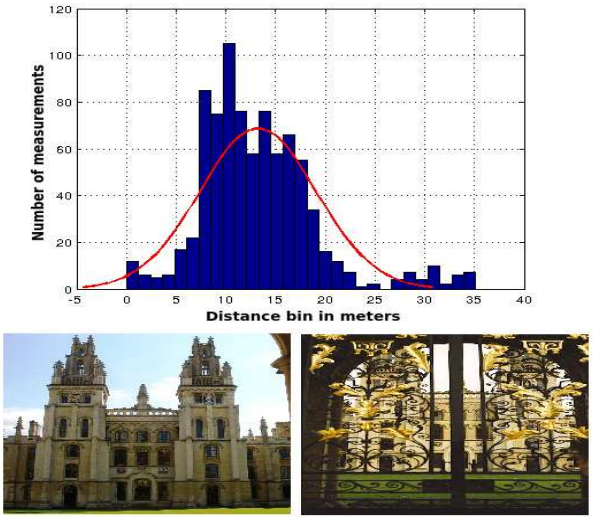


Figure 1: Histogram of GPS localization error (top row) of a *stationary* GPS sensor, showing how inaccurate they can be. Bottom row is an example of two images belonging to the approximately same pose. Visual localization is inaccurate here since in one image the object is occluded, while GPS sensors give accurate localization.

sensors, and satellite communication. Vision based approaches usually provide localization relative to a reference image, and are not global in nature. Visual localization is achieved by matching images using interest points like SIFT, and estimating relative positions by computing and decomposing multiview geometric quantities like the Fundamental/Essential matrix.

There are several advantages and disadvantages of using GPS based localization vis-a-vis visual localization approaches. Commercial viability of GPS sensors make them *cheap* to obtain, thus explaining their ubiquitousness. Such sensors are generally useful for obtaining coarse localization of objects in a *global* coordinate system. They are also not usually affected by the visual quality of an object's surroundings, *i.e.* GPS sensors localize with similar accuracy irrespective of whether they are used on a beach (no unique

978-1-4673-6759-2/15/$31.00 ©2015 IEEE

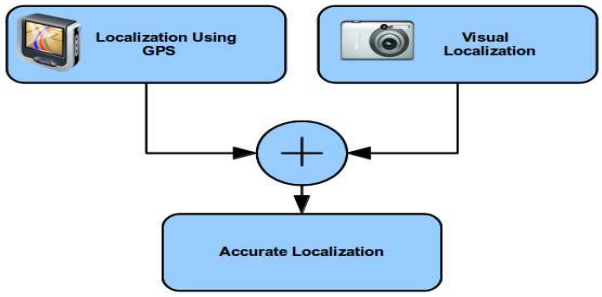
interest points) or near a popular monument (uniquely identifiable structures), and they give *unambiguous* localization to visually similar but differently located places. However, GPS sensors are *inaccurate* beyond a certain point, as illustrated in Figure 1, and can fail in many environments due to reasons such as *sporadic unavailability* of the satellite signal [9]. Thus, *cheap, global, unambiguous, inaccurate, sporadic unavailability* are keywords that characterize GPS sensors.

With the sudden increase of consumer cameras found on portable devices, vision based localization approaches are also now *cheaply* available. Such approaches are generally useful for fine localization of objects in a *local* coordinate system relative to a reference frame. Compared to GPS sensors, vision based localization systems also provide reasonable *accurate* estimates of the object's location. However, chances of *ambiguities* are higher in vision based localization methods since visual similarity of two images of far apart places can lead to erroneous localization estimates. However, since vision based localization methods are not dependent on satellite connectivity, such approaches are readily *available* for utilization. Thus vision based localization approaches can be characterized to be *cheap, local, ambiguous, accurate and available*.

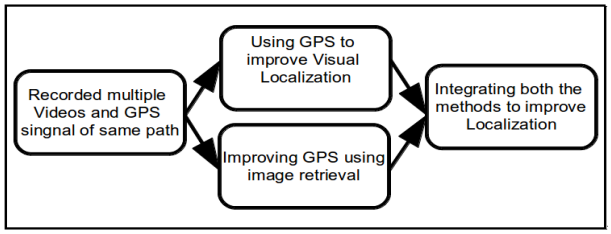
Notice that the two sensors have complimentary advantages and disadvantages. Thus, it is natural to ask *why not combine the advantages of both to improve their accuracy and reliability (Figure 2a)*. With recent portable devices carrying both GPS and vision based sensors, we answer this increasingly important question in this paper. In section 2, we discuss related work. We then describe an approach to improve visual localization using GPS sensory output in section 3. Then we elaborate on an approach to improve GPS output using visual information in section 4, before describing an experiment that complimentarily fuses both the improved estimates to do sequential localization in section 5. Finally we relevant experiments on several datasets to demonstrate the results of our approach in section 6, and conclude in section 7.

2. Related Work

Unreliability in GPS tags critically affects many computer vision tasks like 3D reconstruction and localization for shorter range [8, 3]. This unreliability in GPS has been addressed in several previous works[4]. This includes the use of additional cues such as wifi strength [10], additional special purpose hardware [15] or algorithms that learn error patterns [1]. Vision based methods have been used for GPS tag refinement. Most of the approaches often require a reference point (e.g. Street View) or dataset with pre-assumed correct GPS tag in case of vision based refinement and multi sensor input in case of Kalman filter algorithms [9]. Zamir *et. al.* [13] propose a self-refinement process, that



(a)



(b)

Figure 2: (a) Problem Statement: We want to simultaneously use the noisy GPS signals and erroneous visual localization to generate accurate localization. (b) The block diagram of the proposed method.

has an internal noise reduction and robustness mechanism which effectively uses initial noisy GPS tags of the images to give refined values. Accurate localization is critical to many robotic applications [11, 2].

Visual localization is the problem where the location of a query image is identified by comparison with location-tagged images in a database. Its a challenging problem because images of even the most common scenes like urban environments show wide diversity in appearance. They can vary on different parameters e.g. different viewpoint, scale, occlusion, illumination etc. than the prior images in the database. For this paper we refer occlusion as blocking of the camera view because of non permanent objects. An occlusion in foreground (refer Figure 1) or similar looking images for example, could degrade visual localization. Performance evaluation in visual localization is measured as the Euclidean distance between the GPS tags of query image and retrieved images [11]. Due to inaccuracies in GPS devices people integrate other sensor data like IMU, wheel odometry, and LIDAR sensors [7, 15] to get an accurate localization.

In this paper we propose a method which localizes with an accuracy of 7.5m by fusing vision and GPS together. In this regard, we address 3 main challenges in visual and GPS localization: (i) *Perceptual aliasing*, when similar-looking images have very different GPS locations, (ii) *camera occlusion* (Figure 1), when dissimilar images are co-located, and

(iii) noisy GPS data. To do this, we present an approach to learn the useful feature [14, 6] to improve the localization performance, along with an approach to correct noisy GPS outputs using visual localization [13].

Dataset To do experiments in this paper, we collect 10 video datasets using a Contour action camera, while walking along a 600m path repeatedly over 10 days. We extract images from these videos at `10fps` or `1fps` depending on the requirement. We then extract SIFT features and store them for each frame. While processing 1 video, images from the other 9 are used to build the visual bag of words vocabulary for image retrieval.

3. Use of GPS for Better Visual Localization and Extracting Useful Features

The visual localization problem is often formulated as an image retrieval problem. To achieve this visual features are extracted and clustered to form a visual vocabulary. Bag-of-Words representation models the image database as an unordered set of visual words in the form of an "inverted index". Inverted index is represented as a $(key, value)$ pair where key is the visual word index, and value is the list of the images in which the visual word appears, with their corresponding reliability weights. We identify the visual words in a query image using standard techniques [12]. With the help of visual words and an inverted index, a score of the n^{th} retrieved image is computed as:

$$Score(img_n) = \sum_{z_k \in Z_q} W_k^n \qquad (1)$$

where Z_q is the set of feature descriptors in the query image and W_k^n is the reliability weight of the visual word corresponding to the z_k feature descriptors in the n^{th} image. The image with the highest score is then chosen as the best matching image corresponding to the query image. Due to occlusion many noisy features are extracted that are not useful for the retrieval process. This includes features generated around unstable interest points. We define unstable objects / interest points as those non-stationary objects / interest points that hinder stable retrieval of a given object / location. Our noisy feature rejection module is motivated by the fact that occlusion and unstable object features will likely exist in a single image, while useful features are likely to be found in multiple images of the same object or location. Identifying the features which are robust to change in view can be determined by tracking which features exist in multiple views and are geometrically consistent with one another. This requires a minimum of two views, assuming that the object or location exists in the database prior to the useful feature extraction stage.

(a) (b)

Figure 3: Original image features (a) vs those features which could be considered useful features (b). Transient objects, occlusions in the foreground and non-distinctive areas of the scenes are found to be without useful features.

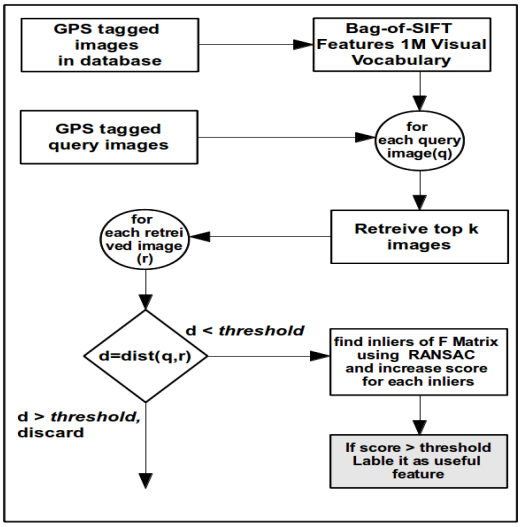


Figure 4: Flow chart for Section 3

Extracting Useful Features and Role of GPS: The process of determining useful image features first starts with the construction of a Bag-of-Words model for an imaeg database. Each image in the database is used as a query and the top k images are then retrieved. All the images in the database have GPS tags associated with them. In order to avoid the perceptual aliasing and camera occlusion we consider only those images which lie within a radius of distance d from the query image. Inliers are then used to estimate epipolar geometry and only features which are geometrically consistent are labelled as useful features. We perform experiments for $d = 10, 20$ and 30 meters and show how it effects visual localization. For details of these experiments please refer sub-section 6.1. A sample result of useful feature extraction is shown in Figure 3. Note that this image

has more than 70% non-distinctive area with occlusion. Our approach has filtered out almost all the noisy features generated around the non-stationary object and non-distinctive area. Figure 4 describes the flow diagram of our proposed method.

4. Improving GPS Signals through Image Retrieval

Zamir *et. al.* [13] proposed a self refinement method to refine user-specified GPS tags of images. We extend their work and apply it to discrete noisy GPS signals in order to get more accurate and consistence GPS signals, which we term *refined* GPS *signals*.

4.1. Details of Algorithm

We have a set S = $\{(V_1, G_1), (V_2, G_2), ..(V_n, G_n)\}$ where V_i and G_i is the i^{th} video and it's corresponding GPS signal. Each GPS signal G_i has a noise attached with them $G_i = \hat{G}_i + \eta$, where $\hat{G}_i$ is the refined signal and η is the noise attached to G_i. Our goal is to extract out $\hat{G}_i$ in a self-refinement manner without using any external reference point.

We sample each V_i at `1fps` and use each frame as query image $\mathcal{I}$ against the rest of the frames from the set $\{$S-$V_i\}$, and retrieve the top μ matches $\{m_1, m_2, ..m_\mu\}$ for each $\mathcal{I}$. We use SIFT Bag-of-Words with vocabulary size of 1 million for the purpose of image retrieval. We then form $\binom{\mu}{2}$ triplets from each query image & each pair of database images and estimate the relative location of the triplet by using Bundler for camera localization [16]. For each triplet $\{\mathcal{I}, m_i, m_j\}$ we get $\{l_\mathcal{I}, l_i, l_j\}$ which are camera locations of $\mathcal{I}, m_i$, and m_j in the SfM local co-ordinate system. However, we note that since in most images the relative height of the camera w.r.t the ground is the same, we could transform these 3D vectors into 2D vectors. Thus, applying an assumption that video tracks were recorded roughly on a planar surface we can reduce the dimensionality of $l_\mathcal{I}, l_i$ and l_j to two (e.g. using PCA).

Our aim is to calculate the GPS of $\mathcal{I}$ using the image triplet $\{l_\mathcal{I}, l_i, l_j\}$. To do this, the locations $\{l_\mathcal{I}, l_i, l_j\}$ should be mapped from SfM local co-ordinate system to the global GPS co-ordinate system. These two Cartesian co-ordinates are related through a similarity transformation matrix RST.

$$\begin{bmatrix} g \\ 1 \end{bmatrix} = (\mathbf{RST}) \begin{bmatrix} l \\ 1 \end{bmatrix} \tag{2}$$

where *l* is a point in the SfM coordinate system and g is it's corresponding point in global GPS co-ordinate system, $\left[\begin{smallmatrix} g \\ 1 \end{smallmatrix}\right]$ and $\left[\begin{smallmatrix} l \\ 1 \end{smallmatrix}\right]$ are homogeneous co-ordinates of g and l. RST is denoted by 3×3 matrix. We need at least two pairs of $g \leftrightarrow l$ correspondence in-order to calculate the RST matrix from Equation 2. In each triplet m_i and m_j are GPS tagged, we use their GPS tags and their locations l_i and l_j to compute RST of the triplet. Now this transformation is used for finding the location of $\mathcal{I}$ in global GPS co-ordinate system. Since we have $\binom{\mu}{2}$ possible triplets, we will get $\binom{\mu}{2}$ possible GPS estimates for a query image.

4.1.1 Robust Estimation through Random Walks

The estimated GPS locations of $\mathcal{I}$ yielded by the triplets is accurate only if the GPS tag of reference images m_i and m_j is accurate. We use Random Walks on estimated triplets to discover the reliable subset of estimations. We define a graph $\mathcal{G}$ = (N,E) where N and E represent the set of node and edges. Each node represents one estimation, i.e. N = $\{g_1, g_2, ...g_\lambda\}$, and there is and edge between each pair of nodes, E = $\{(g_i, g_j), i \neq j\}$. We include the original GPS tag of $\mathcal{I}$, for the estimation of its correct GPS-location, in set N. The transition probability from node i to node j according to their GPS distance is calculated using Equation 3 given by Zamir *et. al.* [13]:

$$p(i,j) = \frac{e^{-\sigma||g_i - g_j||_2}}{\sum_{k=1}^{\lambda} e^{-\sigma||g_i - g_k||_2}} \tag{3}$$

where $||.||_2$ denotes the l_2 norm, λ is the number of node in graph $\mathcal{G}$ and σ is call insensitive parameter.

Image Geo Density and Initial Node Score: The purpose of image geo density is to handle the phenomenon of non uniform distribution of images across popular sites like flickr, facebook, etc ... In our case, where we have multiple videos of the same path, we can assume uniform distribution of images. The initial score of the nodes is not going to be effected by geo-density, hence initial score of the n^{th} node $v(n)$ can be calculated as:

$$v(n) = \frac{1}{\lambda} \tag{4}$$

4.1.2 Adaptive Damping Factor

For a each query image we got a graph $\mathcal{G}$ having λ nodes where each node is initialized with score $v(n)$. The Random Walks algorithm will update the score of one node at every iteration using the transition probability from other nodes to it. Equation 5 is the basic Random Walks formula

$$x_{(k+1)}(j) = \sum_{k=1}^{\lambda} \overbrace{\alpha}^{①} x_k(i) p(i,j) + \overbrace{(1-\alpha)}^{②} v(j) \tag{5}$$

where x_k(i) is the relevance score of i^{th} node at k^{th} iteration. The use of the damping term in Random Walks is to use the prior knowledge about the relevance of nodes and to ensure irreducibility of the transition probabilities matrix which is a convergence condition for Random Walks [5].

The summation of term ① and term ② in Equation 5 should be 1 because the relevance score at any iteration must sum to one, *i.e.*, $\sum_{k=1}^{\lambda} x_k(i) = 1$. Zamir et al. [13] further propose an adaptive damping factor which adaptively changes according to the consistency of each node w.r.t others. They formulate the damping term of a node as a function of its relevance score at each iteration:

$$x_{k+1}(j) = \frac{1}{\eta}\Big(\sum_{i=1}^{\lambda} \overbrace{\big(1-(1-\alpha)x_k(j)\big)}^{①} x_k(i)p(i,j) + \overbrace{(1-\alpha)x_k(j)}^{②} v(j)\Big) \quad (6)$$

The normalization constant η given by Equation 7 forces the sum of all relevance scores to be one.

$$\eta = \sum_{j=1}^{\lambda}\Big(\sum_{i=1}^{\lambda} \big(1-(1-\alpha)x_k(j)\big)x_k(i)p(i,j) + (1-\alpha)x_k(j)v(j)\Big) \quad (7)$$

Estimation of Final GPS-Tag using the Relevance Scores: The estimations which are badly effected by noise are expected to have relevance score of ≈ 0, other nodes should gain the score based on their transition probability and initial score. Finally we compute the refined GPS-Tags of the query $\mathcal{I}$, utilizing a weighted mean using the relevance scores x_π.

$$\hat{g} = \sum_{i=1}^{\lambda} g_i x_\pi(i) \quad (8)$$

where $\hat{g}$ is refined GPS-Tag.

5. Improving the Localization

Integrating Visual Localization and GPS Refinement: In section 3 we have shown how with the help of GPS we can overcome issues like occlusion and perceptual aliasing in order to improve visual localization using image retrieval, whereas section 4 describe how to reduce the error in GPS signal with the help of image retrieval and random walks. As mentioned above, we integrate both modules to improve localization in challenging scenarios like over short distances where GPS signals tend to fail. First we improve GPS signals and label all the images in the database with refined GPS tags. Using these refined GPS tags we filter out the useful features for building a vocabulary and inverted index.

Sequential Localization: Without any loss of generality, we can assume that the motion of any portable device with GPS is a smooth motion. So we can assume such motion satisfies the Markov assumption *i.e.* that is the current pose X_t is dependent only on the previous pose X_{t-1}. Our sequential visual localization method consists of two phases: initial localization phase and query retrieval phase. In initial localization phase we keep on retrieving relevant images to the query image until estimation of the pose is known to us with minimum probability P^*. P^* is calculated using Algorithm 1. Once the initial pose X_t is fixed we use temporal sequence property to fix the pose of X_{t+1} in query retrieval phase. In section 5 we have demonstrated the effect of useful features, refined signal as well as combined effect of both on localization by conducting different set of experiments. We also performed sequential localization after integration.

Algorithm 1 Estimation of initial pose

Input: Bag-of-Word framework, Q: Queue
Output: Initial pose with minimum probability P^*.
for i=1 $\rightarrow N$ **do**
  $r_i \leftarrow$ Retrieved image using query image q_i.
  if Q is NULL **then**
    $Q \leftarrow r_i$
  else
    for j=1 $\rightarrow$ Number of element in Q **do**
      $dist$ = calEculideanDistance(r_i,Q_j)
      if $dist < thresholdDistance$ **then**
        Score(Q_j) = Score(Q_j) + 1
      else
        $Q \leftarrow r_i$
      end if
    end for
  end if
end for
P^* = arg max(Score(p_k) / N ; where $p_k \in$ Q
Return the pose corresponding to p_k

6. Experiments and Results

In this section, we demonstrate the utility of our approach with quantitative experiments. We capture the videos using a Contour action camera with resolution 1920×1080 at $30fps$. The device also has an inbuilt GPS sensor which recorded the corresponding GPS signal at $1Hz$. This gives us two different loosely aligned signals GPS and videos. Since we are also interested in studying the utility of multiple runs captured over time, we collect the videos on ten different days by walking on the same $600m$ long path. Data is captured in the evening at peak traffic hours to ensure approximately same crowded urban environment setting. This process yields ten videos of the same path with labelled GPS tags.

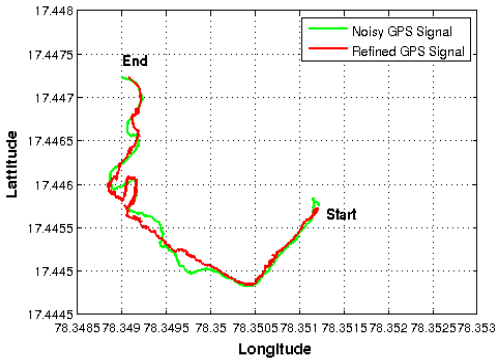


Figure 5: Plot demonstrating the noisy and refined GPS signals. One can observe refined GPS signal (red) is less random and in zig-zag shape as compare to noisy GPS signal (green).

For vision based localization, we use bag of words representation with 1 M visual words built on top of dense SIFT descriptors. During all the experiments, we have ensured that the test data is not used for the vocabulary construction. Wherever, multiple runs are used, one run is used for the evaluation in a leave one out manner.

As can be seen in Figure 5, our sensor fusion scheme yeilds superior estimates of localization. One may notice the noisy (zig zag) path of the original localizations and the more or less smooth path estimated after the multi sensor localization as we discussed in the previous section. In the rest of this section, we demonstrate the quantitative results of various experiments.

6.1. Effect of Useful Features on Visual Localization

First, we demonstrate the utility of feature selection in improving the accuracy of localization. Effect of useful features on visual localization performance was evaluated using K-fold cross validation with number of folds set to 10. For each of the query image we retrieved an image from Bag-of-Words model built by using useful features only. If the Euclidean distance between retrieved image and query image is greater than δ_d we consider the localization to be an "invalid localization". The percentage error in the visual localization is define as:

$$P_e = \left(\frac{TotalNo.OfInvalidLocalization}{No.OfQueryFrames}\right) \times 100 \quad (9)$$

In this experiment we have set the $\delta_d = 10m$. Table 1 shows the comparative percentage error in visual localization between two approaches. In the table experiment E_i means i^{th} video is used as the source for query images and rest all are use in building visual vocabulary. One can observe significant drop in error while using useful features for visual localization. The other observation that can be made is, the drop in percentage error P_e is less for increasing values of d. This is because the number of inliers are almost constant with varying d. Hence we set $d = 10m$ for further experiments to expedite the process without compromising accuracy a lot.

Table 1: Effect of useful features on visual localization. There is a significant drop in percentage error in visual localization with useful features. As d increases the increase in number of inliers are very less. Therefore useful features are almost constant for different values of d. Hence the drop in P_e error is less. NA=Not Applicable, OF=Original Features, UF=Useful Features as shown in Figure 3.

Exp. No.	P_e, *d=NA* *OF*	P_e, *d=10* *UF*	P_e, *d=20* *UF*	P_e, *d=30* *UF*
E_1	31.63	23.56	22.69	22.57
E_2	31.70	23.47	22.65	22.59
E_3	29.49	22.21	21.49	21.52
E_4	32.06	23.93	23.03	22.98
E_5	30.54	22.89	22.15	22.15
E_6	31.98	23.07	22.26	22.23
E_7	31.04	22.87	22.15	22.09
E_8	30.68	22.72	22.13	22.07
E_9	31.75	23.47	22.70	22.58
E_{10}	32.30	23.72	22.63	22.56

6.2. Refined vs Noisy GPS Signals RMSE Score

The recorded GPS signal has noise associated with it. To illustrate with an example, if one were to log the GPS signal while keeping the device stationary, one would see large variation in GPS values of up to 35m (as shown in Figure 1). The term noisy GPS signals refer to the GPS signals which were recorded using GPS receiver. By improving the GPS signals using images (section 4) we get refined GPS signals.

To refine a GPS signal the corresponding video run was used as the query video in the vision based localization. We do this for each of the videos separately. Using all these attempts, we refine the GPS tags and finally we integrate all the GPS tags to get the refined GPS signal.

To quantitatively compare, we used Root Mean Square Estimate(RMSE) values. We assume one of the GPS signal as the ground truth and calculate the RMSE for other signals with respect to assumed ground truth. RMSE for the G_n GPS signal is given as:

$$RMSE(G_n) = \frac{\sqrt[2]{\sum_{i=0}^{N}\left(Dist(G_{gt}(i), G_n(i))\right)^2}}{N} \quad (10)$$

where $N = \min\left(length(G_{gt}), length(G_n)\right)$ as all the signals are approximately same length (so discarding few values will not effect the RMSE much). $G_{gt}(i)$ and $G_n(i)$ is

i^{th} GPS reading of assumed ground truth signal and n^{th} GPS signal respectively. *Dist* calculates the Euclidean distance between two GPS measurements. Mean RMSE of the n^{th} GPS signal is the mean of all the RMSE values taking n^{th} signal as a ground truth. We calculate RMSE values for noisy GPS signals and refined GPS signals separately. Table 2 shows the comparison between Mean RMSE values of noisy and refined GPS signals for each video. S_n indicates that n^{th} signal was taken as ground truth to calculate the Mean RMSE. The Mean RMSE for the noisy GPS signal is ≈10m whereas for refined GPS signals it comes down to ≈6-7m. This demonstrates that the sensor fusion is resulting in a more consistent localization results than using a single sensor alone.

Table 2: Mean RMSE comparison of noisy and refined GPS signal. For refined GPS signal RMSE values ≈7m.

Run No.	*RMSE Noisy Signal*	*RMSE Refined Signal*	*% drop in RMSE Error*
S_1	9.81	6.76	**31.10**
S_2	10.55	7.36	**30.23**
S_3	10.49	7.01	**33.17**
S_4	9.76	6.73	**31.05**
S_5	9.54	6.79	**28.82**
S_6	10.31	6.98	**32.30**
S_7	10.21	6.92	**32.22**
S_8	10.01	6.87	**31.36**
S_9	10.65	7.01	**34.17**
S_{10}	9.96	6.95	**30.22**

6.3. Comparison of Denoising using Synthetic Noise

In another experiment to test the method in various extreme circumstances we added random Gaussian noise as an input error with mean values 100, 500, 1000, 2000, 3000 and 4000 meters to 5, 10, 20, 33 and 50 percent of the 24648 images in our dataset and the standard deviation was set to the half the mean to replicate the GPS device error with $mean = 12.23m$ and $standarddeviation = 5.98$ (Figure 1). We improvised the synthetic error on top of the already existing noise. Hence, the additional synthetic noise determines the lower bound of noise since the exact amount of error in the dataset is unknown. We also made sure that in this experiment, the query images were among the ones with contaminated GPS tags to ensure the evaluation is fair and challenging. We got similar results to the one observed by Zamir *et. al.* (Figure 7.a in [13]). We observed that for a contamination percentages of less than 33%, the method almost completely eliminates the error regardless of the mean of the contamination in the input. Once the input error increase beyond the 33% and 50% the error in output is no more avoidable yet still less than the error in input, which is consistent with the findings of [13].

6.4. Effect of Refined GPS Signal On Localization:

We also performed the experiments to study the effect of refined GPS signal on localization. The ten videos sampled at $10fps$ and tagged each frame with their corresponding noisy and refined GPS tags. Bag-of-Word framework was built using SIFT features from the nine runs with visual vocabulary set to 1 million words. We performed visual localization using one of the video run as a query dataset. Percentage error in visual localization for distance was calculated using Equation 9. Table 3 shows the comparison in visual localization performance between refined and noisy GPS signal for different δ_d distances. An experiment E_i means i^{th} run was used as a query run (each query run contains 4000-4193 frames) and rest all other for vocabulary building. From the the Table 3 one can also infer that for δ_d = 7.5m the difference in the P_e error for noisy GPS signal and refined signal is huge, but as δ_d increases from 10m and beyond P_e is approximately same for both kind of signals. Variance in the P_e for refined GPS signal less than noisy GPS signal as standard deviation are 9.67 and 6.70 respectively for the noisy and refined signal.

Multi Sensor Localization: We integrated both the modules *i) Use of* GPS *for Better Visual Localization* and *ii)Improving the* GPS *Signal Through Image Retrieval* to form a pipeline for more accurate localization. First we refined the GPS signals using images and adjusted the GPS tags therein to the correct locations. Then we improved the visual localization by labelling the useful features with the help of refined GPS . The pipeline was tested on ten video runs, where each runs contains 4000-4193 frames. Nine runs were used to build the Bag-of-Word framework for image retrieval having vocabulary size of 1 million visual words. All the frames from a video run were used as query

Table 3: Effect of refined signal on percentage error. With small value of δ_d there is significant difference in the performance. As δ_d increases the performance gap is narrowed down.NS = Noisy Signal, RS = Refined Signal.

Exp.No.	P_e, δ_d=7.5m		P_e, δ_d=10m		P_e, δ_d=15m	
	NS	RS	NS	RS	NS	RS
E_1	45.98	29.68	31.63	23.11	16.05	15.33
E_2	40.00	33.68	31.73	30.73	18.53	20.00
E_3	32.00	27.85	24.09	24.52	16.19	14.28
E_4	33.00	24.50	25.24	21.28	13.61	10.00
E_5	37.12	29.09	29.43	22.19	16.30	14.80
E_6	39.00	28.67	28.84	25.05	17.46	15.09
E_7	38.85	28.72	28.19	24.23	16.57	14.91
E_8	37.18	28.99	27.79	25.45	16.47	15.00
E_9	40.49	29.81	30.35	23.52	15.82	14.73
E_{10}	36.72	30.01	28.80	24.74	16.14	14.94

Table 4: Comparison of percentage error P_e in localization with δ_d =7.5m while using the methods i)Bag-of-Words framework ii)Bag-of-Words framework + Integration of modules iii)Bag-of-Words framework + Integration of modules + Sequential Localization.

Exp. No.	*Simple BOW*	*BOW Integration*	*BOW Seq. Localization*
E_1	31.63	8.09	2.97
E_2	31.70	8.03	2.73
E_3	29.49	7.10	2.14
E_4	32.06	10.03	3.30
E_5	30.54	7.60	2.71
E_6	31.98	9.07	2.85
E_7	31.04	8.10	2.56
E_8	30.68	7.71	2.60
E_9	31.80	8.29	2.12
E_{10}	31.44	8.56	2.82

dataset. Percentage error in visual localization was calculated by the Equation 9. We localize for the distance δ_d = 7.5m. We also performed the sequential localization by fixing the initial position with Algorithm 1. In the Algorithm 1 we set N = 10($i.e$ re-sampling frequency of the videos) and $thresholdDistance$ = δ_d and P^*=0.7. Table 4 shows significant drop in P_e with multi sensor localization method. Our proposed method is useful to perform localization for a shorter distance($\approx 7.5m$) by fusing images and noisy GPS tags obtained by using any commercial GPS receiver.

7. Discussion and Conclusion

This work proposed a multi sensor based localization scheme that fuse two popular localization schemes to result in a more accurate localization strategy. The objective of our work has been to fuse GPS signals and images together in order to improve the localization. We propose methods that use noisy GPS to improve the vision based localization. We also propose methods that use vision based localization to improve the GPS signals. Finally we use these two steps in an iterative manner to get further improvement. We also presented a set of experiments to validate the fusion schemes and thereby the improvements in localization by fusing image and GPS .

Robustness or accuracy of our proposed method will not be affected by the type of camera, the resolution of the image and occlusion up to a great extent, as our system is built over invariant features like SIFT which take care of issues like change in camera or image resolution. To address the problem of occlusion we use useful features, which handle the occlusion problem pretty well, even when more than half of the image is occluded (Figure 3). Finally, doing multiple runs over the same path is really important, since it is used for building a visual vocabulary and refining the GPS signal.

References

[1] M. Cummins and P. Newman. Fab-map: Probabilistic localization and mapping in the space of appearance. *The International Journal of Robotics Research*, 2008.

[2] M. Cummins and P. Newman. Appearance-only slam at large scale with fab-map 2.0. *The International Journal of Robotics Research*, 2011.

[3] A. Hafez, M. Singh, K. M. Krishna, and C. Jawahar. Visual localization in highly crowded urban environments. In *IROS*. IEEE, 2013.

[4] J. Hays and A. A. Efros. Im2gps: Estimating geographic information from a single image. In *CVPR*. IEEE, 2008.

[5] Y. Jing and S. Baluja. Visualrank: Applying pagerank to large-scale image search. *Pattern Analysis and Machine Intelligence*, 2008.

[6] J. Knopp, J. Sivic, and T. Pajdla. Avoiding confusing features in place recognition. In *ECCV*. Springer, 2010.

[7] J. Levinson, M. Montemerlo, and S. Thrun. Map-based precision vehicle localization in urban environments. In *Robotics: Science and Systems*. Citeseer, 2007.

[8] T.-Y. Lin, S. Belongie, and J. Hays. Cross-view image geolocalization. In *CVPR*. IEEE, 2013.

[9] D. Maier and A. Kleiner. Improved gps sensor model for mobile robots in urban terrain. In *ICRA*. IEEE, 2010.

[10] E. Martin, O. Vinyals, G. Friedland, and R. Bajcsy. Precise indoor localization using smart phones. In *Proceedings of the international conference on Multimedia*. ACM, 2010.

[11] M. J. Milford and G. F. Wyeth. Seqslam: Visual route-based navigation for sunny summer days and dtormy winter nights. In *ICRA*. IEEE, 2012.

[12] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*. IEEE, 2006.

[13] A. Roshan Zamir, S. Ardeshir, and M. Shah. Gps-tag refinement using random walks with an adaptive damping factor. 2014.

[14] P. Turcot and D. G. Lowe. Better matching with fewer features: The selection of useful features in large database recognition problems. In *ICCV Workshops*. IEEE, 2009.

[15] L. Wei, C. Cappelle, Y. Ruichek, and F. Zann. Intelligent vehicle localization in urban environments using ekf-based visual odometry and gps fusion. In *World Congress*, 2011.

[16] C. Wu, S. Agarwal, B. Curless, and S. M. Seitz. Multicore bundle adjustment. In *CVPR*. IEEE, 2011.