Where's Waldo: Matching People in Images of Crowds

Rahul Garg
University of Washington
rahul@cs.washington.edu

Deva Ramanan
University of California at Irvine
dramanan@ics.uci.edu

Steven M. Seitz
University of Washington, Google
seitz@cs.washington.edu

Noah Snavely
Cornell University
snavely@cs.cornell.edu

Abstract

Given a community-contributed set of photos of a crowded public event, this paper addresses the problem of finding all images of each person in the scene. This problem is very challenging due to large changes in camera viewpoints, severe occlusions, low resolution and photos from tens or hundreds of different photographers. Despite these challenges, the problem is made tractable by exploiting a variety of visual and contextual cues – appearance, timestamps, camera pose and co-occurrence of people. This paper demonstrates an approach that integrates these cues to enable high quality person matching in community photo collections downloaded from Flickr.com.

1. Introduction

This work addresses the problem of matching instances of people in images of crowded events. Examples of such events include a football game, a graduation ceremony, weddings, parties, or even popular tourist sites that are photographed many times on the same day. For example, Figure 1 shows several photos from a special event at Trafalgar Square when it was briefly covered with grass. Upon looking very closely, some of the same people can be found to appear in two or more of these images, even though they were taken by four different photographers. Suppose I specify a person in one photo (yellow box, upper left). Can you find her in all of the others? Now suppose that instead of just a few images, there were hundreds or thousands of such photos? This task is akin to the popular Where's Waldo children's book, where the goal is to find Waldo in each image. Applications such as photo browsing and surveillance would immediately benefit from the ability to mine event photo collections for all instances of a person.

This version of Where's Waldo is extremely challenging due to large changes in camera viewpoint, severe occlusions, low resolution and photos from many different photographers – it is truly akin to finding a needle in a haystack. To make the problem tractable, we make the assumption that the rate of photo acquisition is fast compared to the rate of movement of people. Given the exponential growth in the number of photos that people take, the assumption is not unreasonable and will become more and more plausible over time. Further, there are a number of scenarios where people are relatively stationary over large intervals of time (e.g., a football game, a graduation ceremony, etc.). We can then restrict our search for a particular person in a small 3D neighborhood and to photos taken close in time. Thus, this problem becomes a correspondence problem of the form often encountered in vision problems.

Wide baseline matching for rigid, architectural scenes is relatively mature, even at large scale [1, 12]. However, the *people correspondence* problem presents different challenges – people are *nonrigid* objects, who articulate and move over time. Occlusion is severe in crowded scenes. Further, a particular "Waldo" appears in a small fraction of the pictures, as people are dynamic entities occupying the scene for a limited time interval. On the other hand, we exploit the available contextual information (not available in the Where's Waldo books!) to make the problem more tractable. Contemporary image formats contain additional tags such as GPS tags, time stamps. Other forms of context include viewpoint estimation through geometric registration, social context manifested through the co-occurence of friends in each other's photographs, etc. We demonstrate that the task of matching people in crowded events *is* solvable when one exploits these contextual cues.

Our contributions are three-fold. First, we present a novel formulation of people-matching in crowds as a generalization of multi-view stereo, where a part-based appearance classifier is used to score correspondences rather than a simpler pixel or window-based score such as normalized correlation. Second, we show that this matching problem can be significantly aided by the use of contextual cues

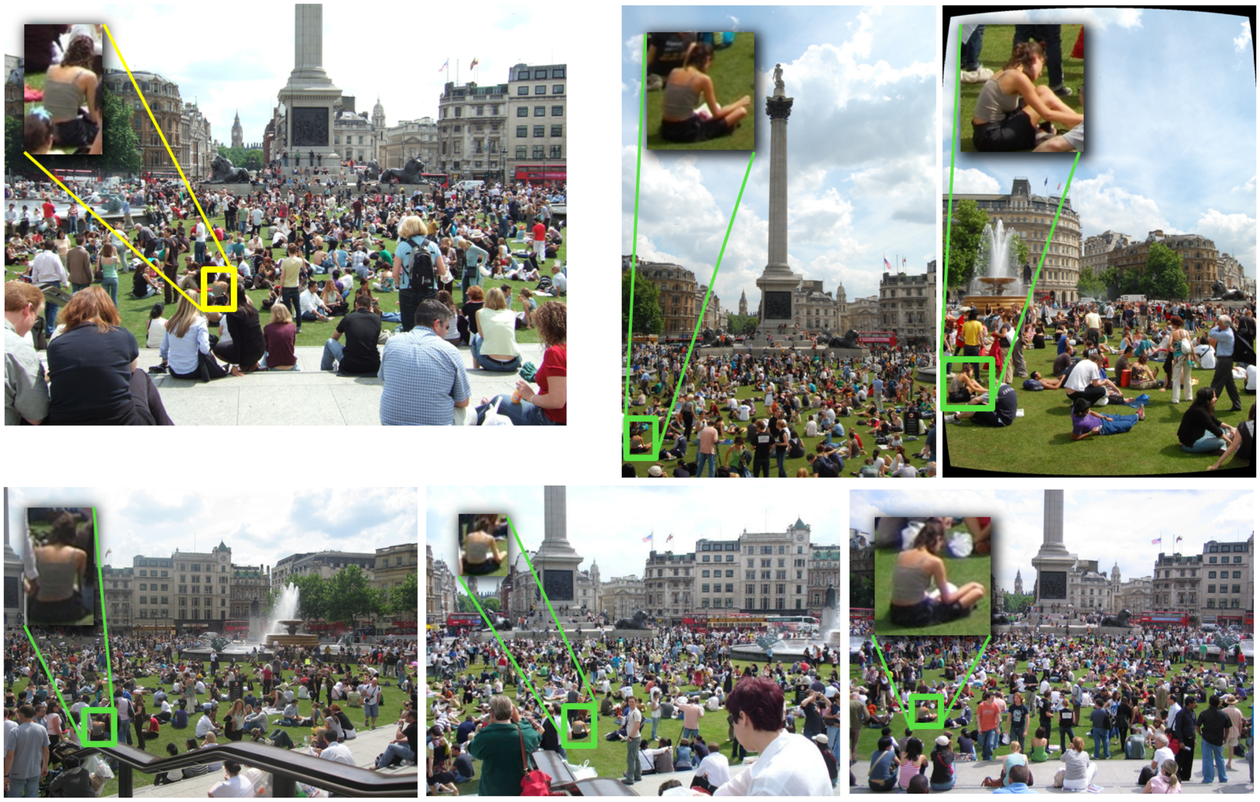


Figure 1: We seek to find all instances of a specific person in a large photo collection. Trained from a single image at the top left, our approach correctly finds 4 of the 5 matches shown above from a collection of 282 images.

(such as co-occurrence and time-stamps) enforced through a novel, global Markov Random Field (MRF) model. Finally, we provide an extensive manually labeled dataset of people matches for benchmarking purposes.

Related work on tagging people in photo collections has focused primarily on cases where face detection and recognition techniques are applicable (i.e., posed, frontal photos) [11, 15, 16] and there are typically only a few people present. In contrast, we seek to find matches in a sea of hundreds of people, and where face detection and recognition methods fail for the vast majority of cases. For example in Figure 1, our final system finds 4 of the 5 matches among which no face is visible at all. We also note that prior authors have explored color models for matching people [7, 11, 13], co-occurrence cues [6], and other contextual cues [8, 13] in other settings.

2. Overview

The input to our system is a collection of photos corresponding to a single event and we aim to find all matches of people marked by the user. We only require a person to be marked in a single image. The user specifies the person by marking different parts (up to 3) in addition to specifying the location of the head and the bottom most point (Section 3.1). A 2D rigid part based color appearance model is learnt from this input (Section 3.2). We register the photo collection using the structure-from-motion system of Snavely *et al.*[12]. We then use the learned appearance model to localize the person in 3D (Section 3.3). Given the location of each person in 3D, we project the location into each image and restrict the search to a small neighborhood (assuming small person movement). Finally, in Section 3.4 we integrate contextual cues (time stamp information, groups of people, etc.) using an MRF framework.

In the paper, we denote a person by p, an image by I and the time stamp of the j^{th} image by t_j. Each person p_i is manually marked in exactly one training image $I_{tr(i)}$.

3. Matching People

3.1. User Input

We require the user to mark a single instance of each person to be searched. The location of a person p_i in an image is specified by clicking on two points in the image: $p_{i_{ground}}$, the point of contact of the person with the ground and $p_{i_{head}}$, the top of the head of the person. In addition, the user specifies different parts (up to 3) of the person by drawing different masks (Figure 2b) which helps build a better

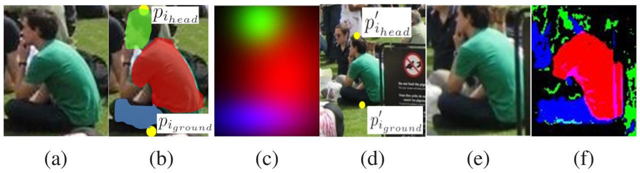


Figure 2: User Input and Appearance Model.

color model as we describe in the next section.

3.2. Learning the Appearance Model

Conventional multi-view stereo methods use a pixel or window-based feature for finding correspondence. Instead, given a training image marked with part masks (Figure 2b), we wish to learn an appearance model for that person which we will use to find correspondences. Building an accurate appearance model is difficult because people can vary greatly in appearance due to changes in viewpoint, scale, occlusions, and exposure/radiometric differences in cameras. We use a part-based appearance model inspired by pictorial structures [4], where parts are restricted to lie in a 2D *rigid* location with respect to a global coordinate frame defined by $p_{i_{ground}}$ and $p_{i_{head}}$.

Color model: For each part, we learn a pixel-level RGB classifier. We tried other features like image patches, SIFT points, etc., but they do not perform well due to limited training data, low resolution and clothes with low texture. Specifically, we create a $9D$ feature x_j for each pixel, consisting of R, G, B values and their quadratic combinations $(RG, R^2, \ldots)$. Labeling pixels inside a part-specific mask as positives $(y_j = 1)$ and those outside as negatives $(y_j = -1)$ (with a 10-pixel band separating them), we learn a logistic regression classifier similar to [9] by computing $w_{part} = \text{argmin}_w \sum_j \log(1 + \exp(-y_j w^T x_j))$. Such a quadratic discriminant can also be obtained by directly estimating a Gaussian model for part pixels and for the background, but we found better results with a discriminative classifier.

Scoring a match: We wish to use the discriminative color models to score a putative match defined by a given candidate $p'_{i_{ground}}$ and $p'_{i_{head}}$ in a new image (Figure 2d). We compute the isotropic scaling, rotation and translation that aligns $p'_{i_{ground}}$ and $p'_{i_{head}}$ with $p_{i_{ground}}$ and $p_{i_{head}}$ respectively and warp the new image according to this transformation (Figure 2e). We then run the part-specific classifiers on the new image to obtain binary classification masks for each of the parts (Figure 2f). Finally, we score the putative match by summing up the number of positively classified pixels inside and immediately surrounding each aligned part. In practice, we use a Gaussian-weighted sum (with Gaussians centered on centroids of the part masks) where pixels inside each aligned part are weighted more heavily (Figure 2c). This also makes the approach less sensitive to the part boundaries input by the user. Also, we surround the Gaussian weights by a ring of negative weights so that blobs of positively classified pixels are scored higher than homogeneous regions.

Occlusions: Parts are often occluded (e.g., the right leg of the person in Figure 2). A simple way to account for occlusions is to define the overall score as the sum of the scores of the individual parts. However, we expect some parts to be more discriminating and reliable for matching. For e.g., a classifier for black hair is not very discriminating. This would suggest a non-uniform weighting of the parts. We experimented with weighing based on the training score but observed that the following approach works well in practice. We simply assume that the first part marked by the user is the most reliable (usually the torso) and constrain it to be visible while we allow for occlusions of other parts. We define the overall score of a putative location as zero if the score corresponding to the first part is zero, otherwise as the sum of the scores of the three parts.

Effectiveness: We found that a globally-aligned, 2D rigid part arrangement sufficed to capture much of the pose variation in our datasets. While such a model is not strictly pose invariant, the parts usually correspond to body parts (e.g., head, shirt, pants) which appear in roughly the same top-to-bottom order in all photos. However, extensions to more flexible deformable models [4] should be straightforward in our framework. We experimented with mixture models as well to model multi-modal color distributions but logisitic regression gave the best results probably due to its discriminative training. We also found the interactive definition of parts to be useful, as oftentimes a user could label multi-colored shirts as multiple parts, which in turn allowed for more accurate appearance models and matches.

3.3. Estimating the 3D Location of a Person

We try to localize the person in $3D$ in a fashion similar to multi view stereo. However, our problem is considerably harder as people are not completely static and appear under different poses (though we still restrict the search to a small 3D neighborhood). Unlike window based or point features, our appearance model is robust to small changes in location. Further, we allow a small amount of *wiggle* when searching for the 3D position, as described below.

The problem amounts to estimating the 3D points $P_{i_{head}}$ and $P_{i_{ground}}$ which project to $p_{i_{head}}$ and $p_{i_{ground}}$ respectively in the training image. For now, assume that the orientation of person in 3D is along the vertical. The vertical direction in the scene can be estimated from a collection of registered photos [14]. Hence, given a candidate 3D location $P_{i_{ground}}$ along the backprojected ray through $p_{i_{ground}}$, $P_{i_{head}}$ is estimated to be the point along the backprojected ray through $p_{i_{head}}$ that lies vertically above $P_{i_{ground}}$ (Fig-

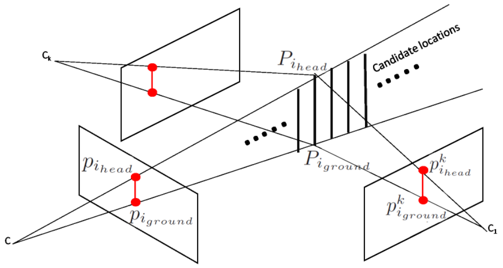


Figure 3: Estimating the 3D location of a person. Given the location of person in an image and assuming that the orientation of the person is vertical, the problem reduces to a 1-D search along the back projected rays.

ure 3). The problem reduces to a 1-D search for $P_{i_{ground}}$ along the back projected ray. We solve it in a fashion similar to multi-view stereo [10], i.e., we exhaustively consider all candidate locations and score each candidate by projecting it into all other images and scoring the projection using the appearance model.

Denote the set of all images by A. Also, denote the projection of a candidate pair $(P_{i_{head}}, P_{i_{ground}})$ into image I_k by ($p^k_{i_{head}}$,$p^k_{i_{ground}}$), which is scored using the appearance model as explained in Sec. 3.2. Denoting the score of this candidate match by $S_i(p^k_{i_{head}}, p^k_{i_{ground}})$, we define the score of the candidate location $(P_{i_{head}}, P_{i_{ground}})$ by $\sum_{I_k \in A} max(S_i(p^k_{i_{head}}, p^k_{i_{ground}}) - thresh, 0)$ where $thresh$ prevents very low scores from contributing. Also, since people tend not to remain perfectly stationary, we allow some slack, i.e., we consider all candidate 3D locations within a small neighborhood of the actual candidate location, and return the maximum score among them. In particular, we consider a window of size $2h \times 2h$ around the projected location where h is the projected height of the candidate location in pixels. Also, for very large collections, we obtained better performance by restricting A to the set of images which have a time stamp close to that of the training image $I_{tr(i)}$.

Height Prior: For each 3D candidate location, we can calculate the 3D height of the person (in scene scale). We therefore impose a prior on the candidate locations based on expected height by multiplying the score obtained in the previous step by $exp(\frac{-(||P_{i_{head}}-P_{i_{ground}}||_2-\mu_h)^2}{2\sigma_h^2})$ where μ_h is the average person height (in scene scale). A crude estimate of μ_h is found by matching a single person manually in two images while a more reliable estimate could be obtained from statistics on the average human height and calibrating the scene. We set $\sigma_h = \frac{5}{3}\mu_h$.

Ground Prior: For scenes where most of the people are sitting on a common ground plane, we constrain $P_{i_{ground}}$ to be close to the ground plane. This is enforced by multiplying the score by $exp(\frac{-(d(P_{i_{ground}}))^2}{2\sigma_g^2})$ where $d(P_{i_{ground}})$ is the distance of point $P_{i_{ground}}$ from the ground plane. We used $\sigma_g = 0.95\mu_h$. The ground plane is estimated by specifying at least three corresponding points on the ground in two images, though it can be automated.

Sensitivity to user input: The algorithm is not very sensitive to user input, particularly the locations of $p_{i_{head}}$ and $p_{i_{ground}}$. Locations of these points determine the similarity transform (scale, rotation and translation) to align the template with the candidate. This transform can be computed correctly if $p_{i_{head}}$ and $p_{i_{ground}}$ are *any* two points in the vicinity of masks as long as they are vertically aligned (which is easy to ensure given the scene vertical). After alignment, actual score is computed via the appearance model which is robust to small localization errors (due to Gaussian weighting). Hence, the actual locations of $p_{i_{head}}$ and $p_{i_{ground}}$ only affects the height and ground priors which are soft priors.

This observation allows us to handle cases when the person is not standing/sitting vertically (e.g., lying on ground). In such a case, we just require the user to enter a point on the ground near the person and point vertically above it roughly at height of the person. While it's possible to use a height prior that allows for both sitting/standing people, we simply use a sitting prior by requiring the user to input $p_{i_{head}}$ near sitting height. Again, any errors in this estimation will only affect the height prior.

3.4. Joint Refinement via MRF Optimization

After previous step, we know the location of each person in 3D. Denote by $S(i, j)$ the appearance model score of person p_i projected into I_j. One can do detection by thresholding $S(i, j)$. However, we also wish to take into account contextual cues, namely

- People tend to appear in same groups, i.e., if a group of people appear together in a few images, they are also likely to appear together in other images as well.
- Images which are nearby in time are likely to contain the same set of people.

Towards this end, we define the *affinity* between pairs of people, $\alpha_p(p_i, p_k)$, and affinity between pairs of images, $\alpha_I(I_j, I_l)$. A higher value of $\alpha_p(p_i, p_k)$ implies that p_i and p_k are likely to appear together. Similarly, a higher value of $\alpha_I(I_j, I_l)$ implies that I_j and I_l are likely to contain the same set of people.

Before we describe how we calculate these affinities, let us see how they are applied. We seek to label each person-image pair (p_i, I_j) as either a positive or a negative detection while taking into account both the appearance model score $S(i, j)$ and the affinity cues.

We model this problem as a Markov Random Field with a node n_{ij} corresponding to every pair (p_i, I_j) over which

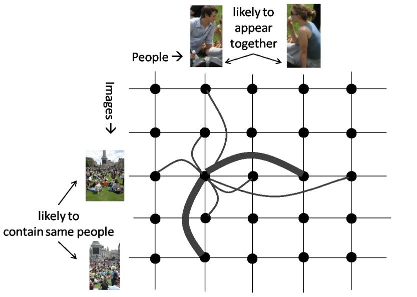


Figure 4: Given the $3D$ location of each person, the problem reduces to deciding whether a person p_i occurs in image I_j which can be visualized as a binary labeling problem over a $2D$ grid. We incorporate grouping priors by adding edges to the graph for pairs of people who are likely to appear together and for pairs of images are likely to contain the same set of people. These edges are shown for a single node in the above figure with weights being proportional to the strength of the priors. We model these correlations via an MRF and solve for the MAP labeling.

we want to compute a binary labeling $\pounds$. If l_{ij} denotes the label of node n_{ij}, $l_{ij} \in \{0, 1\}$ where $l_{ij} = 0$ represents a negative detection and $l_{ij} = 1$ represents a positive detection. Each node is connected to all the other nodes in the same row and column (Figure 4 shows these connections for a single node). The penalty for labeling two nodes differently is defined as

$$P(n_{ij}, n_{i'j'}) = \begin{cases} \alpha_p(p_i, p_{i'}) & \text{if} \quad j = j' \\ \alpha_I(I_j, I_{j'}) & \text{if} \quad i = i' \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

The pairwise potentials in MRF are defined as

$$\phi(l_{ij}, l_{i'j'}) = \begin{cases} 0 & \text{if} \quad l_{ij} = l_{i'j'} \\ P(n_{ij}, n_{i'j'}) & \text{otherwise} \end{cases} \tag{2}$$

In addition to $S(i, j)$, we also compute $R(i, j)$ which is the ratio of $S(i, j)$ to the second highest score in the window which is at least h pixels away from the location with the highest score (h is the projected height of the person). We use the appearance model score $S(i, j)$ and ratios $R(i, j)$ to define the unary potential as follows

$$U(l_{ij}) = \begin{cases} R(i,j)(C - S(i,j)) & \text{if} \quad l_{ij} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where C is a constant that we choose. Intuitively, if $R(i, j)$ is high, we want to weigh the corresponding unary potential more. $R(i, j)$ is clamped above to 20. Similarly, a higher value of C means that a higher $S(i, j)$ is required for a node to be labeled a positive detection.

The nodes corresponding to $(p_i, I_{tr(i)})$ pairs are hard-wired to one. Similarly, the nodes where the appearance model score is zero are hard-wired to zero. Further, if the 3D location of a person falls outside the viewing frustum of an image, or if the projected height of the person is too small, we remove corresponding nodes from the MRF.

The desired labeling is obtained by minimizing the following objective function with respect to the labeling $\pounds$ using Graph Cuts [2]:

$$E(\pounds) = \sum_{ij} U(l_{ij}) + \sum_{ij} \sum_{i'j'} \phi(l_{ij}, l_{i'j'}) \tag{4}$$

We use the MATLAB implementation of Graph Cuts made available by Fulkerson et al. [5]. We also compute the confidence Conf(n_{ij}) of each detection using the following equation which can be computed by running a graph cut for each node [3]:

$$\text{Conf}(n_{ij}) = min_{l_{ij}=0} E(\pounds) - min_{l_{ij}=1} E(\pounds) \tag{5}$$

Computing Affinities: Computing image affinities is straightforward. Images closer in time have higher affinity:

$$\alpha_I(I_j, I_{j'}) = \lambda_1 e^{\frac{-|t_j - t_{j'}|^2}{2\sigma_t^2}} \tag{6}$$

where we used $\sigma_t = 2$ and $\lambda_1 = 0.03$, with time being measured in minutes. Further, we multiply the affinity above by a constant factor if they are taken by the same user (a factor of 4 was found to work well).

We compute $\alpha_p(p_i, p'_i)$ as follows. If D_i denotes the set of images that are known to contain p_i, we define $\alpha_p(p_i, p'_i)$ as

$$\alpha_p(p_i, p_{i'}) = \lambda_2 \frac{|D_i \cap D_{i'}|}{|D_i| + |D_{i'}|} \tag{7}$$

However, we do not know D_i other than the fact that $I_{tr(i)} \in D_i$. Hence we use an iterative approach inspired by EM methods. We initialize $\alpha_p(p_i, p_{i'})$ using the above definition where $D_i = \{I_{tr(i)}\}$. We run the MRF optimization, compute the new detections and then update the affinities according to the new detection and re-run the optimization to get the final detection results. We found that running the MRF optimization 2-3 times while updating affinities is sufficient. Moreover, we keep the constant C used in Eq. (3) high for the first iteration to get a conservative set of detections to estimate $\alpha_p(p_i, p_{i'})$. $\lambda_2 = 0.1$ was found to work well in our experiments.

4. Results

We consider three datasets for evaluation, all downloaded from Flickr.

Dataset 1 contains 34 registered photos taken by a single photographer at Trafalgar Square on May 24th, 2007.

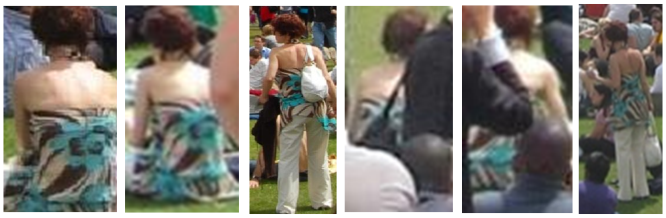


Figure 5: An example set of matches. There are cases with high occlusion and very low resolution.

Dataset 2 contains 282 registered photos of Trafalgar Square taken on May 25^{th}, 2007. These images come from 89 different users and span a larger time window (from morning to evening), making true matches rarer. Figure 1 shows a few typical images from this collection.

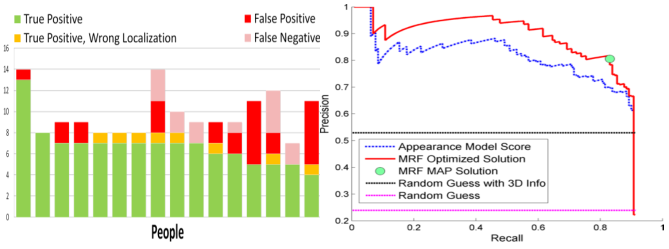
Dataset 3 contains 45 images taken during an indoor event – *HackDay London 2007*. The photos are taken over two days and come from 19 different photographers.

We also used the time-stamps associated with the photos, corrected for timezone offsets by adding the difference between the timezone of the venue (London) with the timezone of the user.

4.1. Preparing Ground Truth Data

To evaluate our results, we manually created a "ground truth" for each dataset. However, finding matches in these photo collections which contains images like those shown in Figure 1 is hard even for humans. Since the photos are registered, we can assist the user in finding matches for the purpose of creating the ground truth dataset. The user starts by marking a person in an image. Then the user is shown all the images one by one with the epipolar lines drawn and he/she only needs to look for a match near the epipolar lines. Once a match is found, the 3D position of the person can be triangulated and the user is then shown the location of the projected points instead of the epipolar lines and he/she can then scan for matches in the neighborhood.

There is a high degree of occlusion in these datasets, but a case is labeled as a positive whenever the human is sure irrespective of the extent of occlusion (Figure 5). Also, while our approach assumes that the people do not move about much, our ground truth includes all matches that the human operator was able to find using our assisted method, including cases where the subject moved outside algorithm's search radius. Such cases are never detected by our algorithm and always count as false negatives. However, we only came across a few such cases implying that they are either rare in these datasets or are extremely hard to spot. In fact, even the assisted matching is quite hard to do manually and our approach sometimes uncovers matches which were missed while preparing the ground truth.



(a) (b)

Figure 6: Dataset 1 (a) Results for individual people (b) Precision-recall curves. In addition to the performance of the appearance model score and the MRF optimized solution, we also show the precision of random guess. The lower horizontal line corresponds to the case when we randomly guess an image to contain a person with a probability equal to the probability of occurrence of true matches. The upper horizontal line shows the performance of random guess with 3D information, i.e., it checks whether the 3D location of the person falls outside the view frustum of the image or if the projection is too small.

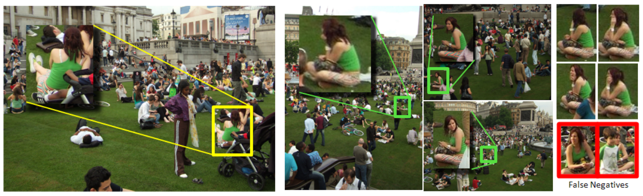


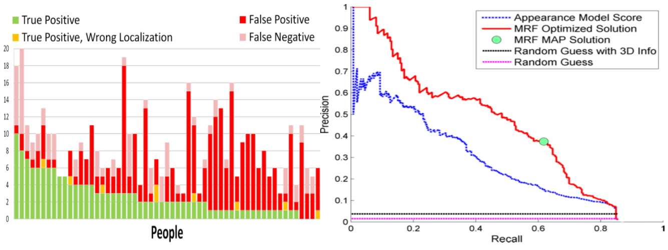
Figure 7: An example where system finds 7 matches for the person on the left all of which are correct. Note that while the training image here was a back pose, all the matches are side poses. The four crops on the right also come for images similar to the three shown. However, there are two missed matches as well (bottom right) which can be attributed to high degree of occlusion and severe pose change.

4.2. Evaluation

The full set of results are provided in the supplementary material. For verification, we consider a detection correct if the distance between the center of the detected location and the center of the true location is less than 0.85 times the height of the person in that image.

Dataset 1 (34 photos): The ground truth had 16 different people and a total of 130 matches. The estimated 3D location is verified by triangulating the ground truth matches (whenever there exists sufficient baseline) and was found to be correct for all people.

Figure 6a shows the results for individual people while 6b shows the precision-recall curves (True positive, wrong location in Figure 6a refers to cases where the image was correctly identified to contain a specific person but the lo-



(a) (b)

Figure 8: Dataset 2 (a) Results for individual people. (b) Precision-recall curves. While the number of false positives may seem high, they only form a very small fraction of the total number of images. The difficulty of this dataset is illustrated by near-zero precision of random guess in contrast with the other datasets.

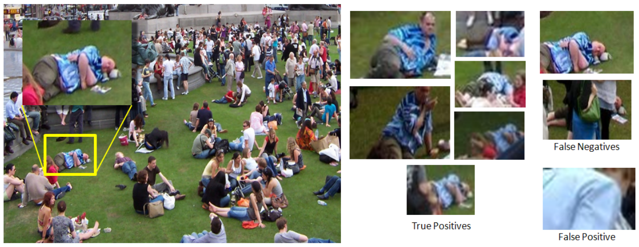
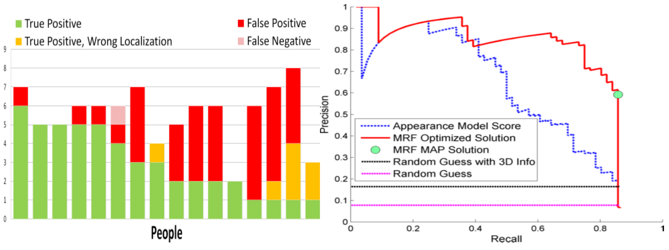


Figure 9: The system retrieves 7 matches for the person marked in the image on the left, 6 of which are correct. One can again see that these are very hard to retrieve due to occlusion, pose changes, illumination changes and low resolution (the sizes of the crops are roughly proportional to the scales at which they were found). One of the missed matches has extreme occlusion. The false positive is due to presence of a similar color.

calization was not correct). The green dot corresponds to the MAP solution while the complete curve for the MRF solution is drawn by using the confidence values from Eq. 5 as scores. To show the improvement, obtained by the contextual cues, we also show the curve corresponding to using the appearance model alone. Precision of random guess is also shown (see Fig. 6 caption for details). Recall remains less than one in the plot as detections with incorrect localization are considered as false negatives irrespective of the threshold. Figure 7 shows an example result from this dataset. The detections include dramatic pose changes and occlusions.

Dataset 2 (282 photos): The ground truth for this particular dataset has 57 people with 244 matches. We purposefully include a few duplicates, i.e., we marked the same person in two different images to evaluate how the choice of training image affects the results. In total, there are 51 unique people.



(a) (b)

Figure 10: Dataset 3: (a) Results for individual people. The last two bars correspond to the cases where the 3D localization failed. (b) Precision-recall curves.

The estimated 3D location was found to be correct for all but 2 queries (which belonged to the same person). However, 6 people in the dataset were located in an elevated part of the scene and hence the ground plane prior had to be turned off for them.

Figure 8a shows results for individual people while Figure 8b shows the precision-recall curves. The number of false positives may seem large but this is a much more challenging dataset as shown by the near-zero performance of the random guess. Contextual cues are especially helpful in a large dataset like this as illustrated by Figure 8b. Figure 9 shows an example result.

Contextual cues encourage people with high affinities to share detections among them. A side effect is that false positives and false negatives are also shared. More user interaction may be helpful here, i.e., correcting a match for a single person may correct it for a number of other people as well. Another side effect of these cues is that they try to hallucinate the person in cases of 100% occlusion, i.e., if a certain set of people are believed to be in a group (have high affinities between them), then the system may try to hallucinate a detection for a certain person if the other people in the group have been detected even if there is little evidence from the appearance model.

For people with duplicate training images, their detections are highly correlated. However, the performance is better when the training image is of higher resolution.

Dataset 3 (45 photos): This dataset is quite different from the other two and is captured indoors. While the matches here are of higher resolution, the problem is made difficult by a lot of people wearing similar clothes. While one is likely to benefit by integrating in face recognition cues in such cases, we demonstrate that our approach still recovers good matches.

The ground truth had 16 people with a total of 56 matches. The 3D location estimation failed for 2 of the 16 people. Both were wearing black clothes, and Figure 12 illustrates why our algorithm fails. However, in spite of the incorrect 3D localization, the contextual cues were able to

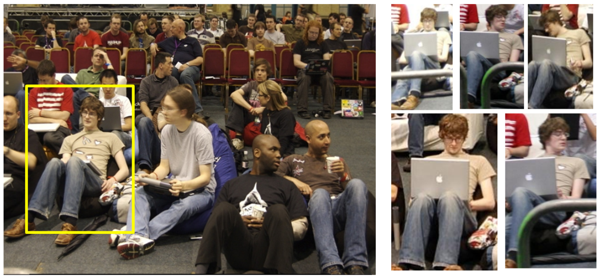


Figure 11: The system finds all 5 matches in this case which include photos from two different photographers. Note that the laptop is not visible in the training image.

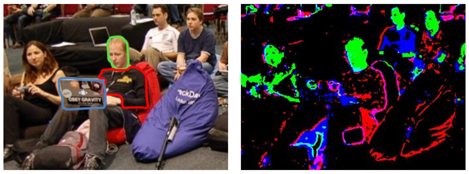


Figure 12: The approach often fails when the person is wearing colors which are common in the scene. The above figure shows the response of the pixel level part detectors on the training image itself. The classification is poor for the red part as the color is not distinct from the background. Also, if there are too many different colors on a single part, the classifier may not be able to find a good discriminating boundary, as is the case for the blue part.

identify the images containing the match. (Figure 10a).

The performance is good on other cases with Figure 11a showing an example. Figures 10a and 10b also reflect this.

5. Conclusion and Future Work

This paper presented an approach for matching people in photos containing hundreds of people, a task difficult even for humans. As future work, we would like to relax the assumptions we make. An important extension would be to allow for large motion, and perhaps the ability to track people's movement through the scene. However, at this point the temporal density of photos is not high enough to do this reliably. More powerful appearance models learned from multiple training images which model humans more accurately would allow one to use larger search neighborhoods. In spite of these assumptions, we have seen that our approach gives good results in a number of challenging and common scenarios and its potential use will continue to grow as the quantity of photo uploads increases.

Acknowledgements: This work was supported in part by National Science Foundation grants IIS-0811878, IIS-0963657 and IIS-0954083, the University of Washington Animation Research Labs, Intel, Microsoft, and Google. We are thankful to Flickr users whose photos we used.

References

[1] S. Agarwal, N. Snavely, I. Simon, S. Seitz, and R. Szeliski. Building Rome in a day. In *Proc. Int. Conf. on Computer Vision*, pages 72–79, 2009.

[2] Y. Boykov, O. Veksler, and R. Zabih. Efficient approximate energy minimization via graph cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(12):1222–1239, November 2001.

[3] C. Desai, D. Ramanan, and C. Fowlkes. Discriminative model for multi class object layout. In *Proc. Int. Conf. on Computer Vision*, October 2009.

[4] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *Int. J. of Computer Vision*, 61(1):55–79, 2005.

[5] B. Fulkerson, A. Vedaldi, and S. Soatto. Class segmentation and object localization with superpixel neighborhoods. In *Proc. Int. Conf. on Computer Vision*, October 2009.

[6] A. Gallagher and T. Chen. Using group prior to identify people in consumer images. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2007.

[7] A. Gallagher and T. Chen. Clothing cosegmentation for recognizing people. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[8] A. Gallagher and T. Chen. Understanding images of groups of people. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, june 2009.

[9] D. Ramanan, D. A. Forsyth, and A. Zisserman. Tracking people by learning their appearance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29:65–81, 2007.

[10] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages I: 519–528, 2006.

[11] J. Sivic, C. L. Zitnick, and R. Szeliski. Finding people in repeated shots of the same scene. In *Proc. British Machine Vision Conference*, 2006.

[12] N. Snavely, S. M. Seitz, and R. Szeliski. Modeling the world from Internet photo collections. *Int. J. of Computer Vision*, 80(2):189–210, November 2008.

[13] B. Suh and B. B. Bederson. Semi-automatic photo annotation strategies using event based clustering and clothing based person recognition. *Interact. Comput.*, 19(4):524–544, 2007.

[14] R. Szeliski. Image alignment and stitching: a tutorial. *Found. Trends. Comput. Graph. Vis.*, 2(1):1–104, 2006.

[15] L. Zhang, Y. Hu, M. Li, W. Ma, and H. Zhang. Efficient propagation for face annotation in family albums. In *MULTIMEDIA '04: Proc. 12th annual ACM Int. Conf. on Multimedia*, New York, NY, USA, 2004. ACM.

[16] W. Zhao, R. Chellappa, P. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Surveys*, 35(4):399–458, 2003.