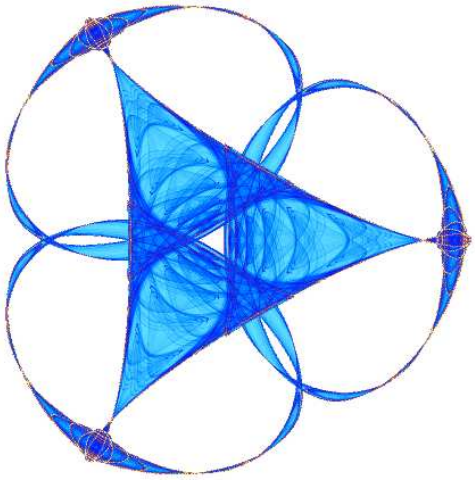
WHAT CAN CASUAL WALKERS TELL US ABOUT A 3D SCENE?

By

Diego Rother
Kedar A. Patwardhan
and
Guillermo Sapiro

IMA Preprint Series # 2173

(August 2007)



INSTITUTE FOR MATHEMATICS AND ITS APPLICATIONS

UNIVERSITY OF MINNESOTA
400 Lind Hall
207 Church Street S.E.
Minneapolis, Minnesota 55455–0436
Phone: 612/624-6066 Fax: 612/626-7370
URL: http://www.ima.umn.edu

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
AUG 2007		**00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
What Can Casual Walkers Tell Us About a 3D Scene? (PREPRINT)	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Minnesota,Institute for Mathematics and its Applications,207 Church Street SE,Minneapolis,MN,55455-0436**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

An approach for incremental learning of a 3D scene from a single static video camera is presented in this paper. In particular, we exploit the presence of casual people walking in the scene to infer relative depth, learn shadows, and segment the critical ground structure. Considering that this type of video data is so ubiquitous, this work provides an important step towards 3D scene analysis from single cameras in readily available ordinary videos and movies. On-line 3D scene learning, as presented here, is very important for applications such as scene analysis, foreground refinement, tracking, biometrics, automated camera collaboration, activity analysis, identification, and real-time computer-graphics applications. The main contributions of this work are then two-fold. First, we use the people in the scene to continuously learn and update the 3D scene parameters using an incremental robust (L1) error minimization. Secondly, models of shadows in the scene are learned using a statistical framework. A symbiotic relationship between the shadow model and the estimated scene geometry is exploited towards incremental mutual improvement. We illustrate the effectiveness of the proposed framework with applications in foreground refinement, automatic segmentation as well as relative depth mapping of the floor/ground, and estimation of 3D trajectories of people in the scene.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**9**	

What Can Casual Walkers Tell Us About A 3D Scene?

Diego Rother, Kedar A. Patwardhan, Guillermo Sapiro
Electrical and Computer Engineering
University of Minnesota, Minneapolis, MN 55455, USA
{diroth,kedar,guille}@umn.edu

Abstract

An approach for incremental learning of a 3D scene from a single static video camera is presented in this paper. In particular, we exploit the presence of casual people walking in the scene to infer relative depth, learn shadows, and segment the critical ground structure. Considering that this type of video data is so ubiquitous, this work provides an important step towards 3D scene analysis from single cameras in readily available ordinary videos and movies. On-line 3D scene learning, as presented here, is very important for applications such as scene analysis, foreground refinement, tracking, biometrics, automated camera collaboration, activity analysis, identification, and real-time computer-graphics applications. The main contributions of this work are then two-fold. First, we use the people in the scene to continuously learn and update the 3D scene parameters using an incremental robust ($\mathbf{L_1}$) error minimization. Secondly, models of shadows in the scene are learned using a statistical framework. A symbiotic relationship between the shadow model and the estimated scene geometry is exploited towards incremental mutual improvement. We illustrate the effectiveness of the proposed framework with applications in foreground refinement, automatic segmentation as well as relative depth mapping of the floor/ground, and estimation of 3D trajectories of people in the scene.

1. Introduction and Motivation

Ordinary movies and footage from static surveillance cameras provide a large amount of video data containing people walking casually in a scene. Processing of such data (for tracking, segmentation, foreground refinement, etc) can benefit tremendously from 3D scene information (even approximate) that can be extracted from such scenes (e.g. see the elegant exposition by Hoiem et al. [3]). These videos, in spite of having a virtually unlimited stream of data, are not commonly used to learn useful scene information apart from the background model. One reason for this is that extracting 3D information about a scene from a single view is an ill-posed problem. To make our goal of online learning the 3D scene plausible, we utilize the information provided by people in the scene.

The authors in [5] and [6] have utilized pedestrians in the scene to estimate the basic 3D scene geometry. The work in [6] introduced camera auto-calibration using the head-foot homology obtained from pedestrians in the scene, with an elegant Bayesian framework. This method requires some prior information about the unknown parameters and also priors about the location of people in the scene. The authors in [5] utilize harmonic homologies as linear constraints on the unknown camera parameters. The head-foot point pairs are collected over time and then processed together to estimate the camera calibration, using a linear Least-Squares type of minimization. Authors in [12] propose a method to estimate camera parameters and light source orientation from 2 views of the scene containing 2 vertical lines and their cast shadows. With a similar aim, but a very different approach, Hoiem et al. [3], learn a graphical model to simultaneously model a scene and put known detected objects (people and cars) in "perspective" in single images.

There have also been a number of interesting recent works, [4, 8, 9, 10], in the area of automatic shadow detection and removal from foreground, which relate to our proposed framework. Most of these works assume that pixels in the shadow have the same chrominance but reduced luminance. Horprasert et al. [4], use such a model to classify a pixel into four categories (including shadows). Recent work in [8] provides a shadow-flow function to detect shadows. This method requires a weak classifier which is initialized as the classifier in [4] involving similar assumptions about shadows. For a detailed review of shadow detection techniques the reader is referred to [9].

The works mentioned above (with the exception of [3]) treat scene geometry and other scene ingredients (like shadows) in complete isolation and therefore cannot benefit from one-another. Techniques like [5] and [6] are non-incremental in nature and therefore are not able to learn or estimate the scene geometry on-line and benefit from the virtually unlimited stream of video data. These approaches

also neglect the effect of shadows and reflections, when timating the head-foot locations, which can lead to err For example, at a certain time of the day, all the peopl the scene cast shadows on the ground plane (usually tected as foreground) in the same direction. If the connecting the head and foot points is computed using second-order moment of the blob as in [6] and [5], t the second moment will have a decided bias towards the rection of the cast shadow, thereby introducing signific errors in the geometry estimation. In our proposed syst we simultaneously learn the scene geometry and shad which are used for mutual refinement, leading to a nu ber of possible applications, some of which are describe Section 6.

In this work, we propose a novel approach for rob incremental and online learning of scene geometry shadow models in such a way that they share a symbi relationship through online mutual refinement. This proach exploits information provided by casual walker the scene, to extract 3D information from the video in s of using a single static camera. We illustrate the effect ness of our framework using real-life applications on c mon videos.

2. Framework Overview and Contributions

An overview of the proposed approach is presented in block diagram of Figure 1. All the blocks in this figure scribe processes that occur online. The videos are assu to be shot from a single static camera during a period of t where no significant illumination changes occurred.[1] first step is to subtract the background in order to anal the information provided by people in the foreground plained in Section 4.1.1). Once the foreground is extrac a rough bounding box (b) is defined around it, as show Figure 2. The next step, computing the head and foot l tions for the current observation (used to improve the c era and shadow model parameters), requires three inp 1) the bounding box b; 2) the previous estimate of the c era and shadow parameters; and 3) the current observa itself. See Section 4 for details. Finally, closing the loop, new estimates for the camera and shadow model parameters are computed from the foot and head locations and the shadow direction just computed (Section 3 and Section 5). This learning process is carried out online and in an incremental fashion. Thus the geometry estimation and shadow model share a symbiotic relationship. The rest of the paper is organized as follows. Section 3, Section 4 and Section 5 briefly describe each of the blocks mentioned above. Section 6 shows the results obtained by the framework and illustrative applications, and Section 7 concludes with a discussion and suggested future work, including preliminary results on reflection modeling for indoor scenes.

[1] Though this issue is not handled here for simplicity, our framework can be easily modified to deal with slow temporal changes.

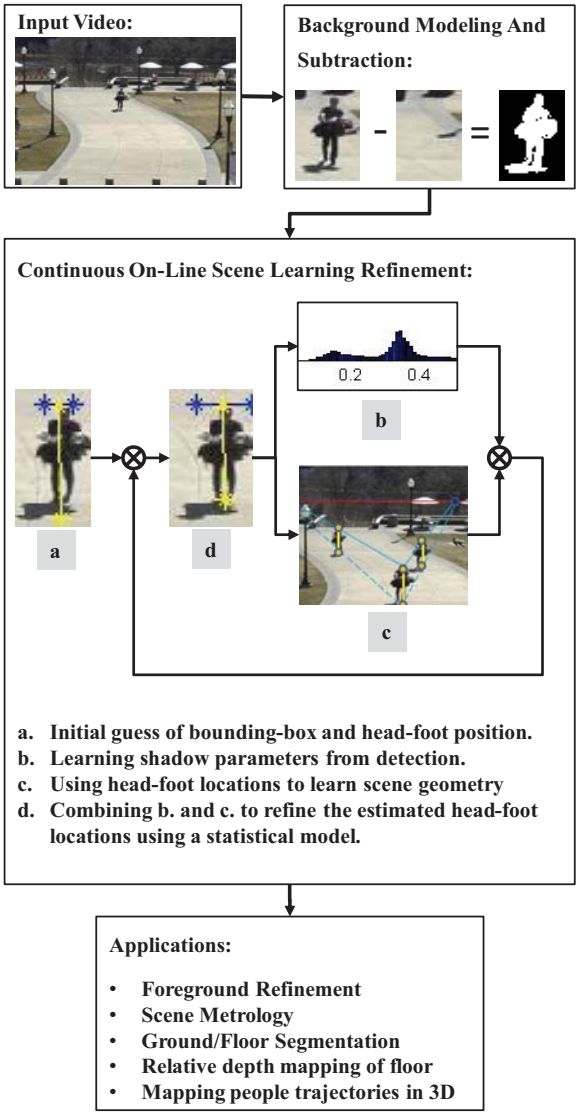


Figure 1. Block diagram of the proposed framework.

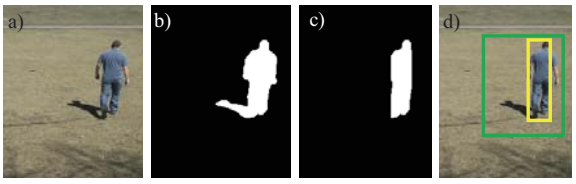


Figure 2. Selection of the initial (rough) bounding boxes. a) Original image. b) Detected foreground. c) Columns having sufficient foreground content. d) Person's bounding box (in yellow) and area to be explained by the model (green box).

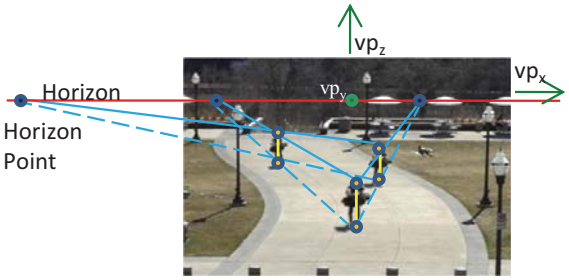


Figure 3. Using people observations for camera calibration. Three different observations of the same person are used to produce three different horizon points (in blue). These, in turn, are used to find the horizon (in red) and the two vanishing points (vp) in the x and y directions (in green).

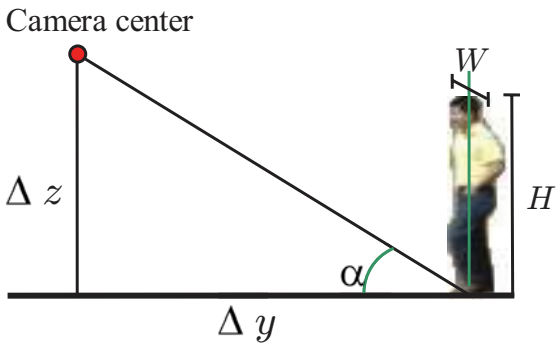


Figure 4. Finding the scale for the y-axis. The ratio of the person's height (H) and width (W) is used to estimate the viewing angle α, and this angle is used to relate the camera height (Δz) to the distance to the person along the y-direction (Δy).

3. Scene Geometry Estimation

A large component of scene understanding is learning the geometric transformation from the image plane to the three dimensional world coordinates. In order to infer true (world) sizes from image measurements, a model for this transformation is necessary. In this work, we make some reasonable, yet simple assumptions regarding the input video (as in [3]). In most videos (surveillance or movies) the camera is positioned in such a way that the people in the scene look upright. Hence we make the assumption that the camera vertical is roughly aligned with the scene vertical. Furthermore, we assume that all those verticals are parallel in the Euclidian sense, implying that the vertical vanishing point lies at infinity. These assumptions are general enough to include most surveillance and still camera videos. Our approach for estimating the scene geometry can be divided into three steps: 1) finding the vanishing line (or horizon) for the floor/ground-plane; 2) composing the camera matrix; and 3) scaling the world coordinate axes. First, to find the horizon, we exploit the homology between the head and foot locations, as suggested by [6]. Assuming the height of a person does not change too much from frame to frame, two observations of the same person provide two parallel horizontal lines that intersect at the horizon (see Figure 3). Each pair of observations (of the same person) then, provides a (noisy) point on the horizon, that we collect in a set H. From the assumption about the vertical vanishing point, the horizon is known to be a horizontal line in the image. We define it to be the horizontal line whose total $\mathbf{L}_1$ distance to the set H of horizon points is minimal. It is well known that the median is the statistic that minimizes this distance. We compute the median online by maintaining a sorted running list of N horizon point observations (e.g. $N = 1000$).[2] Once we have an estimate of the horizon, the second step is to construct the camera matrix. This matrix has, as its first three columns, the image coordinates of the vanishing points of the 3D world axes. The fourth column of this matrix contains the image coordinates of the world origin (see [2] for details). The world origin can be arbitrarily chosen in the floor plane; we choose it to be the center of the visible part below the horizon. The direction of the first two axes can also be arbitrarily chosen by placing their corresponding vanishing points on the horizon line. We chose the y-vanishing point to be the intersection of the middle vertical in the image with the horizon line, and the x-vanishing point to be the point at infinity of the horizon line (due to the symmetry). The z-vanishing point, as mentioned before, is the vertical point at infinity (Figure 3). The third step in the camera calibration procedure is scaling the axes. This is necessary to make approximate inferences about the size of objects or structures in the scene. To scale the axes, the median observed human height and width serve as a reference in the z and x directions respectively. Two different methods were tested to estimate the scaling along the y axis (α_y). The first method relates α_y to the camera height by means of the angle formed by the horizontal and the line connecting the person's feet and the camera center (α in Figure 4). This angle is itself estimated from the ratio between the observed person's height and width. The second method fixes α_y so that the average (ground) speed of the people in the video is 0.5 heights per second. Better results were obtained with the second method (used to generate the results in Section 6), although this scaling constant is admittedly less exact than the other two.

[2] If the number of observations is greater than N we remove equal number of observations from each end of the list to maintain its constant size.

4. Localizing People

Accurately detecting the head and foot locations of the people in the scene is an important ingredient of our approach, since we are using the head-foot locations to estimate the scene geometry (Section 3) as well as estimating the shadow model (Section 5). To correctly locate the feet of

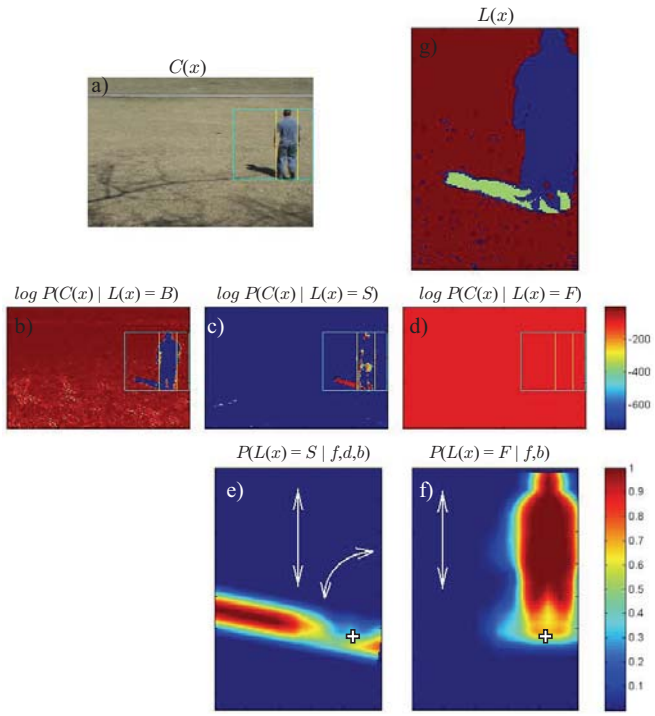


Figure 5. Probabilistic model to localize the person's feet. a) Original image with the person enclosed in the bounding box b (in yellow). The green bounding box contains the data that the model has to "explain". b-d) The log-probabilities that each class assigns to the observed data. e-f) The prior probability of each class. These are functions of the feet position f (marked with a cross), and the shadow direction d. The prior for the background is omitted since it can be computed by imposing that the priors at every pixel add to 1. g) The final labeling of the pixels.

a person in the person's bounding box[3] (shown in yellow in Figure 1d and Figure 5), we design a model to "explain" the set of pixels X observed in the larger bounding box around the person (in green in figures 1d and 5). Figure 5 shows the different parts of this model. We assume that the color $C(x)$ of each pixel x could have been generated by one of three possible classes (or color models), namely, the background, the foreground or the shadow color model. Each pixel x has an associated label, $L(x) \in B, F, S$ indicating the best class assignment for the pixel. Each class defines a distribution in the color space,[4] given by $P(C(x)|L(x))$. Furthermore, each pixel x has a prior probability of belonging to each of these three classes, depending on the feet position f, shadow direction d, and the box b containing the person. This prior is given by $P(L(x)|x, d, f, b)$, described in Section 4.2. Under the usual assumption of independence of the pixels, conditioned on their labels, the likelihood of the observed pixels is given by:

$$P(C(x); L, f, d, b) = \prod_{x \in X} P(C(x)|L(x))P(L(x)|x, d, f, b). \quad (1)$$

To estimate the unknown parameters $L(x)$, f, and d, the expression in Equation 1 has to be maximized. This is done by searching in a grid of (f, d) pairs. Valid directions are restricted to lie in a band around the predicted shadow direction (see Section 5.2). The width of this band is reduced as the confidence in the prediction increases. $L(x)$ is chosen, at each pixel and for each combination of (f, d) values, as the label that gives the highest pixel probability ("best explains" the pixel value). In the rest of this section we explain each of these model parts and the online procedure for learning each one of them.

4.1. Color Models

4.1.1 Background Color Model

As explained above, each class defines a probability density in the color space. The color of each background pixel is modeled by a Gaussian distribution whose mean is the color most frequently seen at that pixel location. The covariance matrix is assumed to be the same for all pixels (captured by a given camera) and is estimated offline from a video of a completely still scene.[5] Figure 5b shows the probability of a sample frame according to this model. More sophisticated models could have been used instead (e.g. [7]), but we found that the model just described is extremely simple, robust, amenable to an online implementation, does not require initial "empty" frames, and works sufficiently well for our purposes.

4.1.2 Shadow Color Model

We assume (as in [3]) that the color of pixels under cast shadows is the original background color attenuated by a global attenuation factor A (which we learn, as explained in Section 5.1), that is dependent only on the ambient lighting conditions at a given time. Thus shadow pixels are modeled using a Gaussian distribution having the same covariance matrix as the background pixels, but the mean of the distribution ($\mu_S(x)$) is the attenuated background color ($\mu_B(x)$) corresponding to the pixel (i.e. $\mu_S(x) = \mu_B(x)A$). Figure 5c shows the probability of a sample frame according to this model.

4.1.3 Foreground Color Model

Having no a priori information about the foreground colors, one way to proceed is to model its color by a uniform distribution (in the color space). This choice is equivalent to set-

[3] We use a simple constraint on the proportions of the bounding box to eliminate non person detections. Since our goal is not to track people, unreliable observations can be discarded without further consequences.

[4] Throughout this work we use the RGB color space.

[5] The covariance was computed only once and works well for all videos.

ting a threshold, and assigning to the foreground class every color that is not well explained by the previous two models. This procedure does not yield good results, because the distribution of foreground colors is not really uniform in real videos. In practice, we observed a wide range of thresholds produced similar results. Consequently we discriminatively selected a threshold and used the same value for all the videos. Figure 5d shows the probability of a sample frame according to this model.

4.2. Location Priors

Given values of f, b, and d,[6] pixels are not equally likely to have been generated by all three color classes. Pixels in a band in the direction d, containing the point f, are obviously more likely to be part of the shadow (Figure 5e). Similarly, pixels above the feet location are likely to belong to the person (foreground). To encode this prior for the foreground, we learned from hand labeled examples, a template in normalized coordinates with respect to the box b (see Figure 5f). This prior was learned offline and only once for all the movies in this paper. For each person detection, this template is stretched vertically to find the feet position that best matches the observation. The background prior (not shown in the figure) is computed so that the three priors add to one.

5. The Shadow Model

Having a good shadow model is key for the accurate location of the person's feet. Learning this model for the shadow implies learning two different components: the luminance attenuation A, produced by the shadow cast on the background pixels, and the vector field of shadow directions on the ground D. As explained in Section 4, each person detection consists of three parameter estimates: 1) the feet location f; 2) the shadow direction d; and 3) a labeling matrix L. These estimates are used to improve the shadow model (both A and D).

5.1. Shadow Attenuation

To learn the luminance attenuation, A, we proceed as follows: 1) for each new detection, collect the attenuations ($\hat{A}(x) = \mu_S(x)/\mu_B(x)$) of the 'Shadow' pixels (according to L); 2) assign a weight to each of these values according to the shadow prior (see Section 4.2) (i.e. pixels in the locations most likely to be shadows are given more importance); 3) keep a histogram of previously observed attenuations and insert the new ones as they arrive; and 4) set A to be the value of the most populated bin (found to be robust enough). The rationale behind this procedure is that only shadow pixels will be concentrated in the histogram. An exception are background pixels that due to noise produce attenuation values close to one (which are ignored). An example of the histogram obtained for the video of Figure 8b is shown in Figure 6.

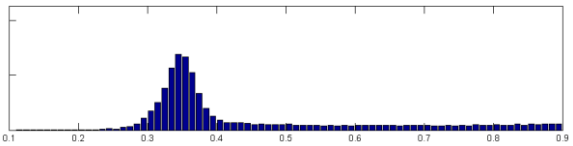


Figure 6. Histogram of shadow attenuations. The position of the maximum is the chosen shadow attenuation.

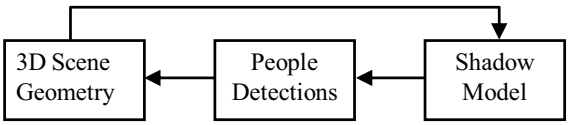


Figure 7. Information flow in the system. The people detections are used to learn the geometry, which in turn is used to learn the shadows, which are used to help in the detections.

5.2. Shadow Direction Field

To learn the shadow direction field D, we define a grid on the floor plane and learn a shadow direction for each grid point. Each new person observation casts a vote for the estimated direction d, at the estimated feet location f. The voting function is given by an anisotropic[7] Gaussian placed at the point f with principal axis d (this is an anisotropic regularization). These Gaussians live on the floor plane which is estimated as part of the scene geometry (Section 3). Votes for directions are accumulated in all the grid points. The direction (at each grid point) that gets the most votes is selected to be the shadow direction at that grid point. This closes the loop for the feedback interaction between all components of the framework (Figure 7). In the next section we show the shadow direction fields obtained for different videos.

6. Results

We now present results of the proposed framework on two different videos shot on two different locations under different illumination conditions (different shadow directions and attenuations). Both videos are of resolution 360x240. These scenes were specifically chosen to lack reflections (that currently our system can not handle) and features that are commonly used to learn the scene geometry (e.g. straight lines, repeating or symmetric structures, etc.). In particular, the video corresponding to Figure 8a has almost no features at all, except for those on the people. Especially challenging is the correct feet localization in the

[6] Recall that f is the foot position, b is the bounding box and d is the shadow direction.

[7] Since the direction of the shadows $D(f)$ changes the least along the direction of the field.

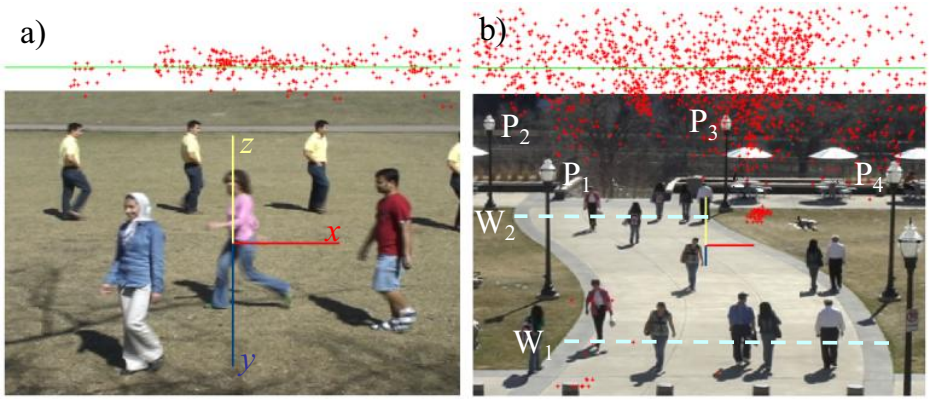


Figure 8. Camera geometry results. The horizon (in green) estimated from the horizon points (a small fraction is shown in red) is used to define a coordinate system in two different videos. The coordinate system is represented by the world origin (on the floor) and one unit vector along each axis: x (in red), y (in blue) and z (in yellow). The unit of measurement is an average person's height. White labels indicate the distances estimated by the framework (see text for details).

Distance	True (m)	Estimated
Lamp post height (P_1)	4.18 ± 0.10	3.94
Lamp post height (P_2)	4.26 ± 0.10	4.38
Lamp post height (P_3)	4.38 ± 0.10	4.39
Lamp post height (P_4)	4.13 ± 0.10	3.92
Path width (W_1)	7.30 ± 0.02	7.53
Path width (W_2)	7.38 ± 0.02	8.31

Figure 9. Comparison of real and estimated measurements in the 3D scene.

video corresponding to Figure 8b, due to the strong shadows in the forward direction whose color is often very close to that of the people in the foreground. Also the camera is quite distant, rendering the people to be as small as 40 pixels. Composite frames from each of these videos are shown in Figure 8a and Figure 8b. In the same figure, the horizon lines and estimated world coordinate systems are also shown. As explained in Section 3, these coordinate systems use the average observed human height as the unit of measurement. Using the metrology techniques in [1] we can contrast ground truth measurements in the scene of Figure 8b against distances estimated from the video, as shown in the table in Figure 9 below. These values were computed assuming an average human height of 1.75m. Please note that this was the only measurement input to the system. No other user input was required in these calculations. Also note that if the shadows were not properly handled, the bias they introduce will render the average human height dependent on the time of the day. Figure 10 shows a shadow detection comparison of our learning based method with that of [4], which is used as a benchmark. We compare with this approach which uses global estimates of shadow attenuation (like we do), unlike [8] which uses separate shadow models for each pixel. Figure 10a and Figure 10b show a qualitative comparison with [4]. Note that we are only using the learned shadow attenuation to compute our shadow region without additional help from the learned shadow direction. If we also use the shadow direction information this shadow detection can be improved further so as to remove the small amount of self-shadows on the person (Figure 10b). To show a quantitative comparison, we manually segmented shadows in 200 video frames (100 each for the two scenes shown in Figure 10). Figure 10c gives a comparative plot of the false alarms generated with our approach (red) against the benchmark (blue). Figure 11 shows the learned shadow directions. The red lines indicate the shadow-field. Observe that the estimated shadow direction field is consistent with the shadows observed in the scene.

Another possible application of the proposed framework is shown in Figure 12. The top row shows, with different colors, the observed trajectories of the people in the scene obtained from the detected feet location (from the camera perspective). From each pixel in the image, we compute a weighted distance[8] (second row) to the trajectories, as in [11]. A simple thresholding of this distance is used to segment the floor (third row). Note that the floor segmented in this way includes only pixels that have been "stepped on" (and pixels close to them in both location and color), unlike what would be obtained if all foreground pixels were included. This segmentation evolves as more trajectories are observed. Figure 13 shows two of these trajectories on the ground/floor, inversely warped using the estimated camera transformation. These trajectories can be verified from the uploaded videos. Notice that the person on the right is a biker, who was still correctly exploited by the system towards the estimation of the scene geometry.

[8]The weight is given by the magnitude of the image gradient.

7. Discussion and Future Directions

In this paper we demonstrated the robust extraction of useful 3D information from ordinary scenes by exploiting information provided by people in the foreground and the large amount of such observations that a video contains. 3D characteristics of the scene like its geometry and shadow model are estimated in a mutually beneficial fashion, which leads to a number of useful applications such as accurate foreground detection; relative depth mapping of the ground plane along with the trajectories of the people on the ground, leading to better analysis of the scene; and scene metrology.

The automatic detection and joint modeling of reflections (especially in indoor scenes) and shadows is an important future direction of work, which as shown in the case of shadows, will also have a symbiotic relationship with the geometry estimates. Contrary to shadows, the reflection location is completely specified once the camera geometry is known. Models could exploit this fact, together with the shadow information, to locate the feet position more accurately. We show preliminary results in this direction in Figure 14. This type of on-line joint modeling of 3D scenes from single views will also allow for real-time person insertion in live videos (along with their shadows and reflections). Results in these directions will be reported elsewhere.

Acknowledgments: Work partially supported by ONR, NSF, DARPA, NGA, and ARO.

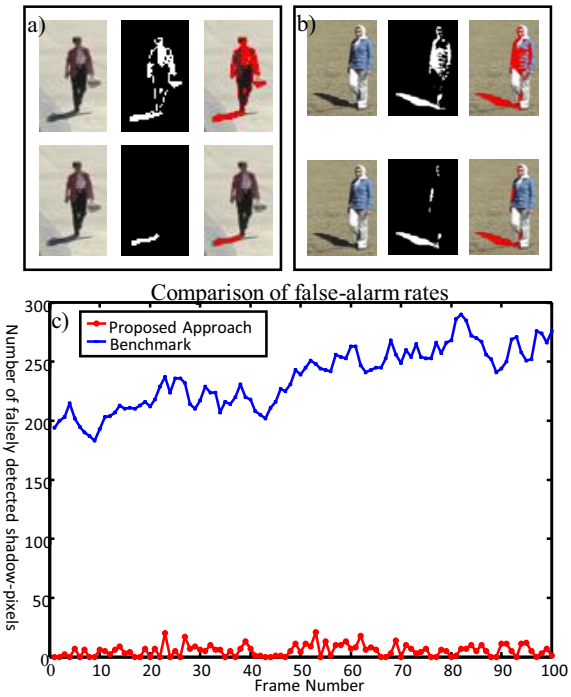


Figure 10. Shadow detection for foreground refinement. The top rows in figure (a) and (b) show shadow detection results from [4], with shadow mask (center) and shadow region indicated in red (right). The bottom rows in (a) and (b) are results from our proposed online shadow modeling approach. The plot in (c) compares the false-alarms for the video in (a) produced by [4] and this work, using a set of manually segmented shadows as ground-truth (see videos uploaded).

References

[1] Criminisi, A.: Accurate Visual Metrology from Single and Multiple Images. Springer Verlag (2001)

[2] Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2000)

[3] Hoiem, D., Efros, A., Herbert, M.: Putting Objects In Perspective. Proc. IEEE CVPR **2** (2006) 2137–2144

[4] Horprasert, T., Harwood, D., Davis, L.: A Statistical Approach for Real-time Robust Background Subtraction and.. IEEE ICCV Frame Rate Workshop (1999)

[5] Junejo, I., Foroosh, H.: Robust Auto-Calibration From Pedestrians. Proc. IEEE ICVSS (2006)

[6] Krahnstoever, N., Mendonça, R.S.: Bayesian Autocalibration For Surveillance. Proc. IEEE ICCV (2005) 1858–1865

[7] Mittal, A., Paragios, N.: Motion-based Background Subtraction Using Adaptive Kernel Density Estimation. Proc. IEEE CVPR (2004) **2** 302–309

[8] Porikli, F., Thornton, J.: Shadow Flow: A Recursive Method to Learn Moving Cast Shadows. Proc. IEEE ICCV (2005) 891–898

[9] Prati, A., Mikic, I., Trivedi, M., Cuchiara, R.: Detecting moving shadows: Algorithms and evaluation. IEEE Trans. Pattern Analysis And Machine Intillegence (2003) **25** 918–923

[10] Salvador, E., Cavallaro, A., Ebhrahimi, T.: Shadow Identification and Classification Using Invariant Color Models. Proc. IEEE ICASSP (2001) 1545–1548

[11] Yatziv, L., Bartesaghi, A., Sapiro, G.: $O(N)$ implementation of the fast marching algorithm. Journal of Computational Physics, Academic Press Professional, Inc, (March 2006) **212:2** 393–399

[12] Cao, X., Shah, M.: Camera Calibration and Light Source Estimation from Images With Shadows. Proc. IEEE CVPR **2** (2005) 918–923

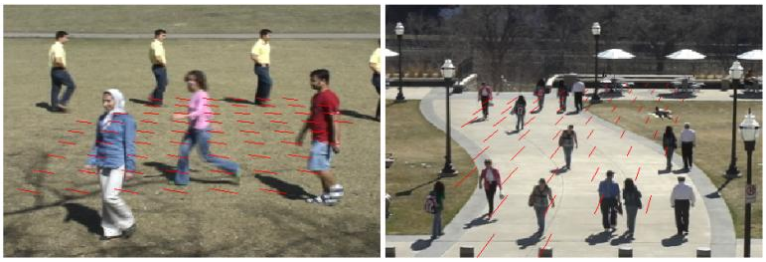


Figure 11. Shadow direction fields computed for two different videos. The red lines represent the shadow directions in the floor points where the lines start.

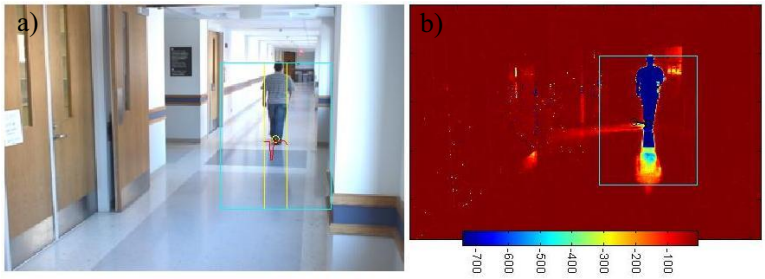


Figure 14. A challenging scene containing shadows and reflections. a) Original image. The yellow circle shows the detected feet position. b) Background Log-Probability. Note that part of the reflection of the person in the floor is as unlikely background as the person itself, but the feet position is still correctly located.

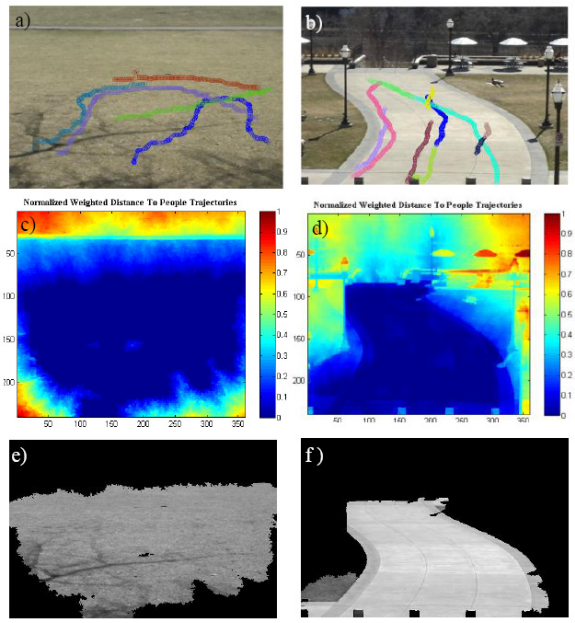


Figure 12. Floor segmentation. a-b) Observed people tracks as computed from the detected feet position. c-d) Weighted distance to the tracks (see text for details). e-f) Segmented

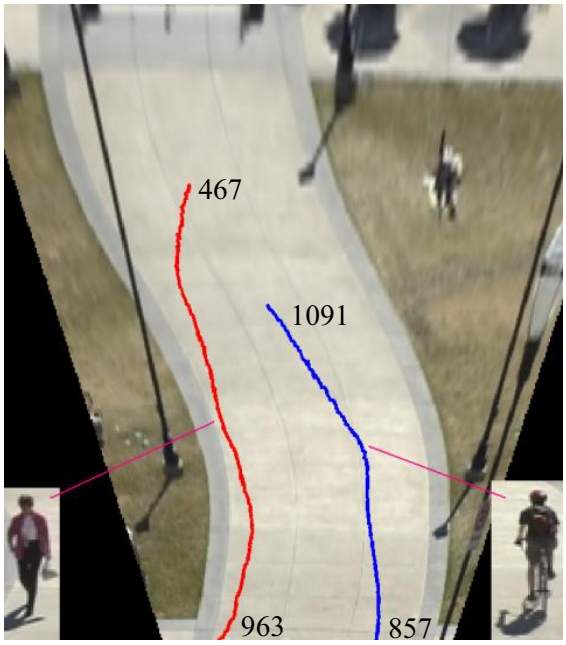


Figure 13. Aerial view of the tracks of two people. Note that one of them is a biker. The estimated camera matrix was used in the projective transformation to warp the floor to its true size. Frame numbers (relative to the video uploaded as supplemental material) are shown next to the tracks to allow the verification of the trajectories.