SPATIOTEMPORAL INPAINTING FOR RECOVERING TEXTURE MAPS OF PARTIALLY OCCLUDED BUILDING FACADES

Christopher Rasmussen

Dept. Computer & Information Sciences
University of Delaware
cer@cis.udel.edu

Thommen Korah

Dept. Computer & Information Sciences
University of Delaware
korah@cis.udel.edu

ABSTRACT

We present a technique for constructing a "clean" texture map of a partially occluded building facade from a series of images taken from a moving camera. Building regions blocked by trees, signs, people, and other foreground objects in a minority of views can be recovered via temporal median filtering on a registered image mosaic of the planar facade. However, when such areas are occluded in the majority of camera views, appearance information from other visible portions of the facade provides a critical cue to correctly complete the mosaic. In this paper we apply a robust measure of spread to infer whether a particular mosaic pixel is occluded in a majority of views, and introduce a novel spatiotemporal *timeline-based* inpainting algorithm that uses an additional motion cue in order to fill the texture map in majority-occluded regions.

1. INTRODUCTION

As part of a vision-based architectural modeling project (see [1, 2] for related work by others), we want to capture the visual appearance of buildings via robot-based "scanning." Given a polyhedral model of a building's structure, a major subgoal of the task is to obtain a high-fidelity texture map or *elevation* of each planar section of its facade. There are numerous issues related to motion planning and exploiting positional sensors for this problem, but here we focus only on some of the key computer vision and image processing issues that arise. Given a sequence of overlapping images of a single large plane of a building wall that have been taken with a general goal of good coverage, we aim to reconstruct an accurate map of that section of the facade.

Creating a planar mosaic via homography estimation has been thoroughly studied [3, 4, 5]. The complicating factor that motivates this paper is the possible presence of other, unknown objects in the scene between the camera and building plane—e.g., trees, people, signs, poles, and other clutter of urban environments. Without explicitly recognizing them, these objects may be erroneously included in the building appearance model. Assuming that the building plane accounts for the majority of pixels in the sequence, with robust methods we can estimate the dominant motion of the building and stabilize it against the camera motion. This converts the problem of "occluder removal" to a background subtraction problem—or rather its corollaries *foreground subtraction* [6, 7, 8] and *layer extraction* [9, 10, 11, 12]. Many of these approaches either assume that the moving objects are relatively small compared to the background, facilitating temporal median filtering [6, 7], or that the objects to be removed are manually identified once [13, 8] in order to segment them later.

Image/video inpainting [14, 15, 16, 17], a method for image restoration or object removal, offers a way to remove larger foreground elements. Typically, the region to be filled is user-specified, but in this work automatically-identified occluded regions serve as the areas to be filled. This is strictly necessary only where the background is never seen for the entire sequence, but our chief innovation is using regions visible in at least one view to constrain what should be painted there. By combining spatial information from pixels in a partially-completed mosaic with the temporal cues provided by images in the *timeline*, or sequence of images captured, sequences that present significant difficulties for temporal median filtering can be well-handled. In the sections that follow, we will detail our techniques for image registration, estimation of foreground likelihoods over the timeline, and integration of this information into a spatiotemporal inpainting algorithm built upon the non-parametric method described by Criminisi *et al.* in [15].

2. METHODS

2.1. Image registration

We begin by computing the dominant planar motion (assumed to belong to the building facade) between successive pairs of images $\mathbf{I}_t, \mathbf{I}_{t+1}$ in a sequence of N frames. These initial frame-to-frame homographies $\mathbf{H}^*_{t,t+1}$ are computed by matching KLT features [18] in both frames followed by RANSAC for outlier rejection [19] . Taking frame number $ref = \lceil \frac{N}{2} \rceil$ of the sequence as the *mosaic reference frame*, the homographies are then concatenated together to align each frame with the mosaic—i.e., $\mathbf{H}^*_{ref,ref}$ is the identity; for $t < ref$, $\mathbf{H}^*_{t,ref} = \mathbf{H}^*_{ref-1,ref} \cdots \mathbf{H}^*_{t+1,t+2}\mathbf{H}^*_{t,t+1}$; and similarly for $t > ref$. Warping each frame $\mathbf{I}_t$ by $\mathbf{H}^*_{t,ref}$ with bilinear interpolation results in a mosaic-aligned frame $\mathbf{W}^*_t$.

Computing frame-to-mosaic homographies this way worsens misalignment errors for frames distant from the reference. With additional constraints on frame alignments (e.g., that the first and last or other temporally distant image pairs overlap), global consistency methods [20] or other forms of *bundle adjustment* may mitigate such errors. Currently, we assume a 1-D scanning motion around the building perimeter and thus cannot take advantage of these methods. Thus, we minimize alignment errors by refining the initial feature-based homographies with a robust direct method that iteratively minimizes the sum of squared differences (SSD) between frames [21, 22]. This procedure operates sequentially on adjacent pairs of warped images $\mathbf{W}^*_i, \mathbf{W}^*_j$ starting from $\mathbf{W}^*_{ref}$ and working outward. After concatenating these refined pairwise homographies, we obtain a final set of refined frame-to-mosaic homographies $\mathbf{H}_{t,ref}$ and stabilized images $\mathbf{W}_t$.

2.2. Identifying Problem Pixels

Each location $\mathbf{p} = (x, y)$ in the mosaic reference frame has a set of pixels from the warped images $\{\mathbf{W}_t(\mathbf{p})\}$ associated with it which we call its *timeline* $\mathcal{T}(\mathbf{p})$. The size of each timeline $|\mathcal{T}(\mathbf{p})|$ may vary from 0 to N depending whether the pixel at $\mathbf{p}$ was imaged or not in each frame. Intuitively, since all pixels on the building facade exhibit the dominant motion, they should appear stationary in the mosaic whereas foreground objects such as trees and signs move due to parallax. Given that each $\mathcal{T}(\mathbf{p})$ contains an unknown mixture of background and foreground object pixels, our goal is to correctly pick or estimate each background pixel $\mathbf{M}(\mathbf{p})$ where $|\mathcal{T}(\mathbf{p})| > 0$, forming a building mosaic $\mathbf{M}$. In this paper we assume that the lateral and vertical limits of the building associated with corners, the roofline, the ground, etc. are given, and we do not rectify the mosaic to compensate for an oblique viewing angle.

A robust estimator for $\mathbf{M}(\mathbf{p})$ under the assumption that foreground pixels are in the minority (i.e., outliers) in $\mathcal{T}(\mathbf{p})$ is the *temporal median* $\mathbf{M}(\mathbf{p}) = \mathit{median}(\mathcal{T}(\mathbf{p}))$. This is computed separately for each color channel: $\mathbf{M}_{red}(\mathbf{p}) = \mathit{median}(\mathcal{T}_{red}(\mathbf{p}))$, and so on, giving rise to the median mosaic $\mathbf{M}_{med}$. This estimator fails, however, when foreground pixels are in the majority in a particular timeline. We observe that except for large homogeneous foreground regions or *camouflaged* foreground objects with almost the same color as the background, the likelihood that $\mathcal{T}(\mathbf{p})$ has a majority of foreground pixels is proportional to the variability or "spread" of its color distribution. To robustly measure this variability, we use the median absolute deviation (MAD) [23], defined as $\mathit{MAD}(\mathcal{T}(\mathbf{p})) = \mathit{median}(|\mathbf{W}_t(\mathbf{p}) - \mathit{median}(\mathcal{T}(\mathbf{p}))|)$ over all t in the timeline. A scalar MAD value is obtained at each pixel by computing it separately for each color channel and summing. A high MAD value at $\mathbf{p}$ indicates a higher likelihood that $\mathbf{M}_{med}(\mathbf{p})$ is unreliable, so unreliable median mosaic pixels are filtered out by thresholding their MADs—these are so-called *MAD outlier* pixels. Finally, the raw MAD outlier mask is spatially smoothed with a morphological majority operation.

2.3. Inpainting Missing Pixels

In this section we present an algorithm for filling the MAD outliers in $\mathbf{M}_{med}$ that is built upon the work in Criminisi, Pérez, and Toyama [15], a patch-based copying method combining ideas from non-parametric texture synthesis and diffusion-based inpainting. We will refer to their method as *CPT inpainting*.

2.3.1. Review of CPT inpainting

As diagrammed in Fig. 1, an empty target region Ω's pixels are filled from its border $d\Omega$ inward by copying square image patches from a source region Φ to target patches $\Psi_{\mathbf{p}}$ centered on $\mathbf{p} = (x, y) \in d\Omega$. Given the next target patch $\Psi_{\hat{\mathbf{p}}}$, an *exemplar* patch $\Psi_{\hat{\mathbf{q}}}$ is selected from Φ and pixels are copied to the unfilled portion of the target patch $\Psi_{\hat{\mathbf{p}}} \cap \Omega$ from the corresponding part of $\Psi_{\hat{\mathbf{q}}}$. Letting the entire image region be denoted by $\mathcal{I}$, $\Psi_{\hat{\mathbf{q}}}$ is chosen as the source patch with the minimum SSD between it and the already-filled part of the target patch $\Psi_{\hat{\mathbf{p}}} \cap (\mathcal{I} - \Omega)$ (normalized for area). As inpainting proceeds Ω shrinks while Φ remains constant, leaving a band of filled pixels $\Omega_0 - \Omega_t$ at step t. Note that Φ can be smaller than $\mathcal{I} - \Omega_0$.

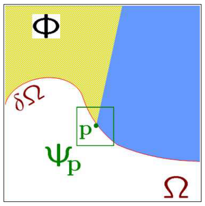


Fig. 1. Source region Φ, target region Ω, target boundary $d\Omega$, target patch $\Psi_{\mathbf{p}}$ (from Criminisi *et al.* [15])

A priority function $P(\mathbf{p}) = C(\mathbf{p})D(\mathbf{p})$ sets the order in which patches along $d\Omega$ are filled. $C(\mathbf{p})$ is a *confidence* term that measures the amount of reliable information around $\mathbf{p}$ with the formula $\sum_{\mathbf{q} \in \Psi_{\mathbf{p}} \cap (\mathcal{I} - \Omega)} C(\mathbf{q})/|\Psi_{\mathbf{p}}|$. Initially, $C(\mathbf{p}) = 0 \; \forall \mathbf{p} \in \Omega_0$ and $C(\mathbf{p}) = 1 \; \forall \mathbf{p} \in \mathcal{I} - \Omega_0$. When pixels in $\Psi_{\hat{\mathbf{p}}} \cap \Omega$ are filled in, their confidence values are updated from 0 to $C(\hat{\mathbf{p}})$, having the effect of preferring sections of $d\Omega$ that were filled earlier vs. later.

$D(\mathbf{p})$ is a *data* term proportional to the dot product of the tangent vector to $d\Omega$ at $\mathbf{p}$ and the gradient vector $\nabla_{\mathbf{p}}$ with the maximum magnitude in $\Psi_{\mathbf{p}} \cap (\mathcal{I} - \Omega)$. This encourages the extension of linear structures by boosting the priorities of patches with a strong edge "flowing into" them—as, for example, in Fig. 1.

2.3.2. Timeline Inpainting

Let the MAD outlier pixels be the target region Ω and the rest of the median mosaic $\mathbf{M}_{med}$ be the source region Φ. Our problem differs from pure spatial inpainting in that the timeline $\mathcal{T}$ for each $\mathbf{p} \in \Omega$, provided it contains at least one background pixel, should constrain the filling process. Thus, our major goals are to determine which, if any, pixels in $\mathcal{T}(\mathbf{p})$ are from the building background, and to integrate this information into the inpainting process. Letting $\mathcal{T}(\Psi_{\mathbf{p}}) = \{\Psi_{\mathbf{p}}^1, \dots, \Psi_{\mathbf{p}}^{|\mathcal{T}(\mathbf{p})|}\}$ be the timeline of patches centered on $\mathbf{p}$, we create a *timeline mosaic* $\mathbf{M}_{time}$ by modifying CPT inpainting in three major ways:

1. In the first of two stages, each patch-wise pixel copy to Ω comes *from one timeline patch* $\Psi_{\hat{\mathbf{p}}}^* \in \mathcal{T}(\Psi_{\hat{\mathbf{p}}})$ maximally likely to have come from the building
2. During stage one, the updated confidences $C(\mathbf{p})$ of newly-filled pixels are set to the motion-based *background likelihoods* $p_{back}^*(\mathbf{p})$ of the pixels in $\Psi_{\hat{\mathbf{p}}}^*$
3. If the mean background likelihood $\bar{p}_{back}(\Psi_{\hat{\mathbf{p}}}^t)$ for every patch in $\mathcal{T}(\Psi_{\hat{\mathbf{p}}})$ is below a threshold τ_{back}, $\Psi_{\hat{\mathbf{p}}}$ is *not filled* at that time. Stage two begins when all remaining areas of Ω meet this definition, and consists simply of CPT inpainting

Each of these three modifications is explained below:

Timeline patch selection Consider a patch $\Psi_{\hat{\mathbf{p}}}$ in the mosaic $\mathbf{M}_{time}$ that is the next to be inpainted. Pixels in its unfilled part $\Psi_{\hat{\mathbf{p}}} \cap \Omega$ will come from the corresponding part of one timeline patch $\Psi_{\hat{\mathbf{p}}}^* \cap \Omega$. We copy pixels from the timeline rather than Φ to maximize correctness, improve feature alignment, and allow for the retention of unique features not present in Φ. To pick a $\Psi_{\hat{\mathbf{p}}}^*$ that is most likely to contain building pixels rather than foreground pixels, we rely upon two cues: (1) Appearance-based similarity to other features in the presumed "all-building" region Φ; and (2) Minimal motion energy (indicating no occlusion in that frame).

Most buildings have repeated patterns such as windows, doors, columns, bricks, etc., so building (as opposed to foreground) timeline patches in Ω are likely to have a similar appearance to features in Φ. However, SSD-based appearance matching alone is a less reliable indicator of "buildingness" in homogeneous areas, and can be improved by incorporating the likelihood that motion occurred in that patch in a particular timeline frame. By combining the unfilled portions of each timeline patch with the filled part from the mosaic to create a timeline of *composite patches* $\mathcal{T}(\tilde{\Psi}_{\hat{\mathbf{p}}}) = \{(\Psi_{\hat{\mathbf{p}}}^t \cap \Omega) \cup (\Psi_{\hat{\mathbf{p}}} \cap (\mathcal{I} - \Omega))\}$, we jointly measure patch t's building similarity and motion energy with the formula $B(\tilde{\Psi}_{\hat{\mathbf{p}}}^t) = \min_{\mathbf{q} \in \mathcal{I}} |\tilde{\Psi}_{\hat{\mathbf{p}}}^t - \Psi_{\mathbf{q}}|^2 / \bar{p}_{back}(\Psi_{\hat{\mathbf{p}}}^t)$,[1] with $\Psi_{\hat{\mathbf{p}}}^*$ determined by $* = \mathrm{argmin}_t \, B(\tilde{\Psi}_{\hat{\mathbf{p}}}^t)$.

[1] $\forall \mathbf{q} \ni$ the area fraction $f = A(\Psi_{\mathbf{q}} \cap \Phi)/A(\Psi_{\mathbf{q}}) > 0.75$. The SSD $|\cdot|^2$ is computed over source pixels in $\Psi_{\mathbf{q}} \cap \Phi$, and is normalized by $1/f$

The intersection of a pair of successive, thresholded difference images was suggested in [24] as a method for identifying foreground pixels. By converting the warped images to grayscale and scaling their intensity values to $[0, 1]$ to get $\{\mathbf{W}'_t\}$, we can adapt this approach to define a motion energy or *foreground image* at time t as $\mathbf{F}_t = (|\mathbf{W}'_t - \mathbf{W}'_{t-1}|) \otimes (|\mathbf{W}'_{t+1} - \mathbf{W}'_t|)$ where $| \cdot |$ is the absolute value and $\otimes$ is the pixelwise product.[2] Letting μ be the mean foreground image value over all t, we define the *background likelihood* for pixel $\mathbf{p}$ in warped image t as $p^t_{back}(\mathbf{p}) = e^{-\mathbf{F}_t(\mathbf{p})/\mu}$, and $\bar{p}_{back}(\Psi^t_{\hat{\mathbf{p}}})$ as the mean pixelwise background likelihood over all pixels in $\Psi^t_{\hat{\mathbf{p}}} \cap \Omega$.

Confidence term The background likelihoods $p^*_{back}(\Psi_{\hat{\mathbf{p}}} \cap \Omega)$ are copied as the confidence values of the newly filled-in pixels in $\Psi_{\hat{\mathbf{p}}} \cap \Omega$. This tends to limit the propagation of bad choices in subsequent iterations—i.e., patches bordering areas of higher motion energy are bypassed for low motion energy areas first. The decaying confidence scheme of CPT inpainting does not apply because timeline patch pixels in the interior of Ω are no less reliable than those near its edges.

Stopping criterion With no patch in $\mathcal{T}(\Psi_{\hat{\mathbf{p}}})$ from the background, there are no temporal constraints on what pixels to fill it with. Because unique features in Ω may not be similar to any patches in Φ, we detect all-foreground timelines solely on the basis of excessive motion energy. Specifically, if for every patch in $\mathcal{T}(\Psi_{\hat{\mathbf{p}}})$ the mean background likelihood $\bar{p}_{back}(\Psi^t_{\hat{\mathbf{p}}}) < \tau_{back}$, $\Psi_{\hat{\mathbf{p}}}$ is not filled. Subsequent inpainting in adjacent areas may allow some skipped pixels to be filled later, but stage one halts when this condition is true at every remaining $\mathbf{p} \in \Omega$. The holes that are left are generally much smaller than Ω_0, with more building structure revealed, and thus stage two can consist of pure CPT inpainting with much better results than if it had been run in place of stage one.

3. RESULTS

In the limited space available here, we describe the operation of our algorithm on a single image sequence. 801 24-bit color frames, resampled to 360×240 pixels each, were captured at 30 fps from a camera moving parallel to a building facade. Several objects at different depths occlude parts of the building including trees, bushes, and a large sign. Our algorithm was run on a subset of 17 frames from the sequence taken at intervals of every 50 frames; four examples of these are shown in Fig. 2.

The median mosaic $\mathbf{M}_{med}$ shown in Fig. 3(a) is mostly quite good, recovering almost all of the facade cleanly (radial distortion was automatically removed from the input frames using the method in [19]). The near tree (e.g., frame 750 in Fig. 2) is almost entirely removed (some artifacts near the mosaic edges are due to an insufficient number of overlapping images there). This is because its large parallax motion causes occlusions to be brief and thus tree pixels are in the minority in the timeline vs. building pixels. A significant problem area, however, zoomed in Fig. 4, is created by the more distant tree, which exhibits relatively little parallax motion. This object occludes many building pixels in a majority of frames, confounding the median filter as shown in Fig. 4(a).

Areas where $\mathbf{M}_{med}$ is poor correlate well with the MAD outliers. The result of a conservative threshold which tags about 20% of pixels as outliers is shown in Fig. 3(b). CPT inpainting to fill Ω_0 is insufficient, as too much structure is hidden. With a 21×21 patch size and a search region Φ of all MAD inliers (above the manually-chosen ground plane border indicated by the red line in Fig. 3(b)), Fig. 4(b) shows the central window behind the small tree replaced by a second-story doorway (!).

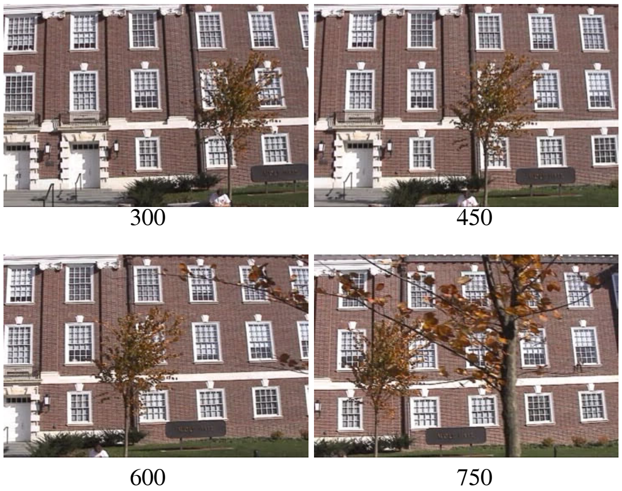

300
450
600
750

Fig. 2. Raw frames from building sequence

A temporal alternative to $\mathbf{M}_{med}$ is the pixel-wise maximum likelihood (ML) method of copying the pixel from the timeline which has the lowest motion energy. Results for $\mathbf{M}_{ML}$ around the window are shown in Fig. 4(c). This method does not cause blurring as $\mathbf{M}_{med}$ does, and often succeeds with the background visible in only a minority of the timeline. However, when there is *no* background visible anywhere in the timeline, a foreground pixel is erroneously drawn.

The results after stage one of timeline inpainting are shown in Fig. 3(c) and Fig. 4(d). For this stage, a shifting, circular search region $\Phi(\mathbf{p})$ (radius $= 150$ pixels[3]) around each 11×11 patch's center $\mathbf{p}$ was used. In Fig. 4(d) it appears that the unfilled pixels after stage one ($\tau_{back} = 0.6$) are correlated with the areas where $\mathbf{M}_{ML}$ is incorrect. The results after CPT inpainting in stage two are shown in Fig. 3(d) (post-processed with automatic affine rectification based on vanishing point identification) and Fig. 4(e).

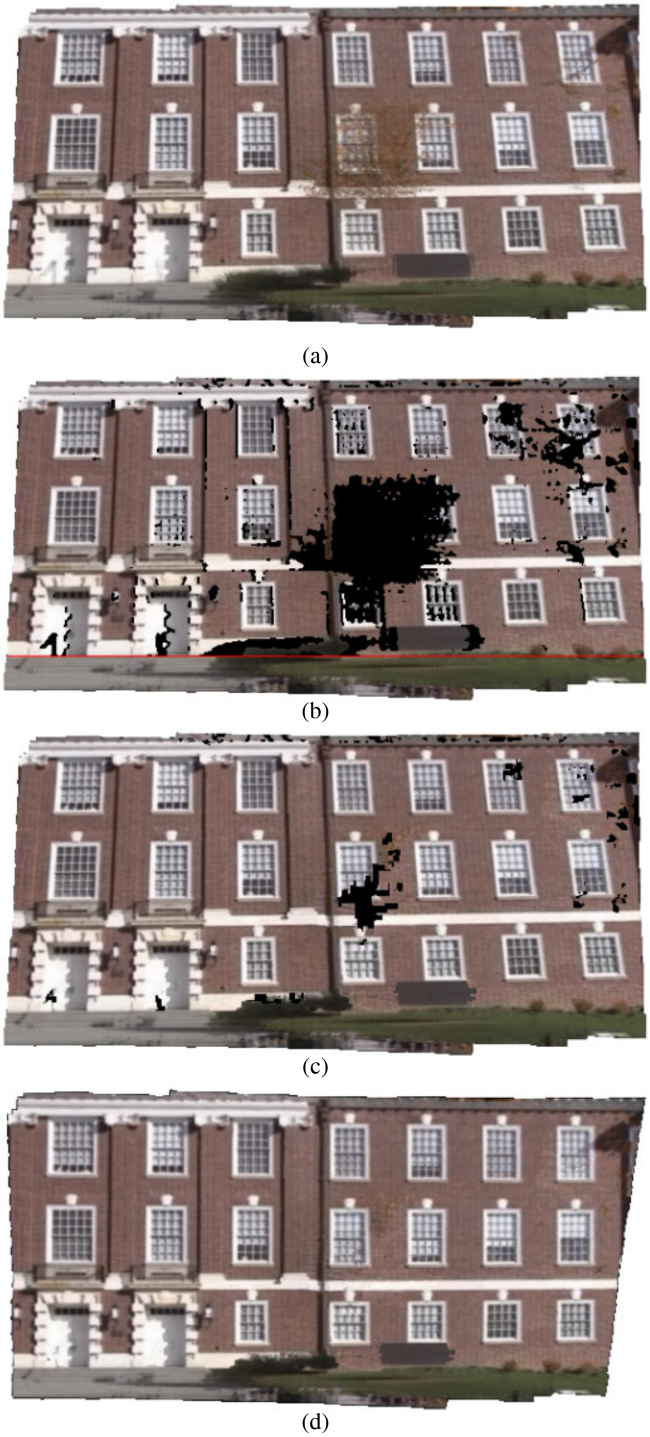
4. CONCLUSION

We have presented a novel approach to detecting and removing occlusions of building facades in image sequences using a combination of temporal and spatial inpainting. An important unaddressed image processing issue is the identification of homogeneous regions which are foreground in every frame of the sequence, which MAD does not detect. Higher-level pattern recognition—such as classification to differentiate the largely vertical and horizontal textures of buildings from the organic patterns of trees, for example—will likely be necessary for this step, followed by pure spatial inpainting (perhaps multi-scale) to fill in missing pixels.

As part of our larger architectural modeling project, we are currently investigating techniques for straight line analysis and vanishing point detection [2] to automatically rectify the texture map and segment planar facade regions from the ground and each other at building corners.

[2] This of course excludes the timeline's first and last images

[3] For reference, window centers are about 65 pixels apart vertically and horizontally



(a)

(b)

(c)

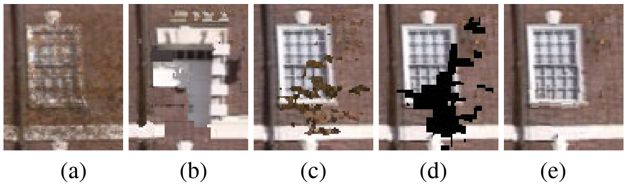
(d)

Fig. 3. (a) Median mosaic $\mathbf{M}_{med}$; (b) MAD outliers in $\mathbf{M}_{med}$; (c) $\mathbf{M}_{time}$ after stage one; (d) $\mathbf{M}_{time}$ after stage two (rectified)

.

5. REFERENCES

[1] S. Teller, M. Antone, Z. Bodnar, M. Bosse, S. Coorg, M. Jethwa, and N. Master, "Calibrated, registered images of an extended urban area," *Int. J. Computer Vision*, 2003.



(a) (b) (c) (d) (e)

Fig. 4. Comparison of solutions in small tree problem area. (a) $\mathbf{M}_{med}$; (b) CPT inpainting on MAD outliers; (c) $\mathbf{M}_{ML}$: timeline pixels with maximum background probabilities; (d) Timeline inpainting after stage one; (e) $\mathbf{M}_{time}$ after stage two

[2] F. van den Heuvel, *Automation in Architectural Photogrammetry; Line-Photogrammetry for the Reconstruction from Single and Multiple Images*, Ph.D. thesis, Delft University of Technology, Delft, The Netherlands, 2003.

[3] J. Davis, "Mosaics of scenes with moving objects," in *Proc. IEEE Conf. Computer Vision & Pattern Recognition*, 1998.

[4] M. Hansen, P. Anandan, K. Dana, G. van der Wal, and P. Burt, "Real-time scene stabilization and mosaic construction," in *DARPA Image Understanding Workshop*, 1994.

[5] R. Szeliski, "Video mosaics for virtual environments," *IEEE Computer Graphics & Applications*, vol. 16, no. 2, 1996.

[6] D. Farin, P. de With, and W. Effelsberg, "Robust background estimation for complex video sequences," in *Proc. IEEE Int. Conf. on Image Processing*, 1997.

[7] F. Odone, A. Fusiello, and E. Trucco, "Layered representation of a video shot with mosaicing," *Pattern Analysis & Applications*, vol. 5, pp. 296–305, 2002.

[8] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut - interactive foreground extraction using iterated graph cuts," in *SIGGRAPH*, 2004.

[9] J. Wang and E. Adelson, "Representing moving images with layers," *IEEE Trans. Image Processing*, vol. 3, no. 5, 1994.

[10] A. Fitzgibbon, Y. Wexler, and A. Zisserman, "Image-based rendering using image-based priors," in *Proc. Int. Conf. Computer Vision*, 2003.

[11] J. Xiao and M. Shah, "Motion layer extraction in the presence of occlusion using graph cut," in *Proc. IEEE Conf. Computer Vision & Pattern Recognition*, 2004.

[12] N. Jojic and B. Frey, "Learning flexible sprites in video layers," in *Proc. IEEE Conf. Computer Vision & Pattern Recognition*, 2001.

[13] A. Kokaram, B. Collis, and S. Robinson, "A bayesian framework for recursive object removal in movie post-production," in *Proc. IEEE Int. Conf. on Image Processing*, 2003.

[14] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *SIGGRAPH*, 2000, pp. 417–424.

[15] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Processing*, vol. 13, no. 9, 2004.

[16] J. Jia, T. Wu, Y. Tai, and C. Tang, "Video repairing: Inference of foreground and background under severe occlusion," in *Proc. IEEE Conf. Computer Vision & Pattern Recognition*, 2004.

[17] Y. Wexler, E. Shechtman, and M. Irani, "Space-time video completion," in *Proc. IEEE Conf. Computer Vision & Pattern Recognition*, 2003.

[18] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Conf. Computer Vision & Pattern Recognition*, 1994.

[19] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.

[20] H. Sawhney, S. Hsu, and R. Kumar, "Robust video mosaicing through topology inference and local to global alignment," in *Proc. European Conf. Computer Vision*, 1998.

[21] M. Irani, B. Rousso, and S. Peleg, "Computing occluding and transparent motions," *Int. J. Computer Vision*, 1994.

[22] M. Black and P. Anandan, "The robust estimation of multiple motions: parametric and piecewise-smooth flow fields," *Computer Vision & Image Understanding*, 1996.

[23] T. Tommasini, A. Fusiello, E. Trucco, and V. Roberto, "Making good features to track better," in *Proc. IEEE Conf. Computer Vision & Pattern Recognition*, 1998, pp. 178–183.

[24] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers:, "Principles and practice of background maintenance," in *Proc. Int. Conf. Computer Vision*, 1999.