Interactive Techniques for Registering Images to Digital Terrain and Building Models

Billy Chen[1] *Gonzalo Ramos*[2] *Eyal Ofek*[1]
Michael Cohen[3] *Steven Drucker*[2] *David Nistér*[2]

[1]Virtual Earth, [2]Live Labs, [3]Microsoft Research
Microsoft
One Microsoft Way
Redmond, WA 98052
{bill.chen, gonzalo, eyalofek}@microsoft.com
{mcohen, sdrucker, dnister}@microsoft.com

ABSTRACT

We investigate two interactive techniques for registering an image to 3D digital terrain and building models. Registering an image enables a variety of applications, including slideshows with context, automatic annotation, and photo enhancement. To perform the registration, we investigate two modes of interaction. In the *overlay* interface, an image is displayed over a 3D view and a user manually aligns 3D points to points in the image. In the *split* interface, the image and the 3D view are displayed side-by-side and the user indicates matching points across the two views. Our user study suggests that the overlay interface is more engaging than split, but is less accurate in registration. We then show several applications that make use of the registration data.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

General terms: Design, Experimentation

Keywords: registration, calibration.

INTRODUCTION

There is currently a huge proliferation in the number of digital pictures that are taken every day. Location tags can play an important role in helping us recall and organize our photos. Mentioning a place reminds us of our experiences there, including what we felt and sensed. This sensory information, in particular, what we perceived, helps us to recall our photos. For example, thinking of Paris may remind us of those photos taken on the perfect day atop the Eiffel tower, or the photos of friends and family at the Champ de Mars during a beautiful summer sunset. Location information can be collected by the cameras themselves, as many have now added GPS hardware which records the global coordinates where a picture was taken. People can also annotate photos using interactive geo-tagging tools. This kind of location information can now be accessed by many popular photo-sharing websites for tours and exploration.

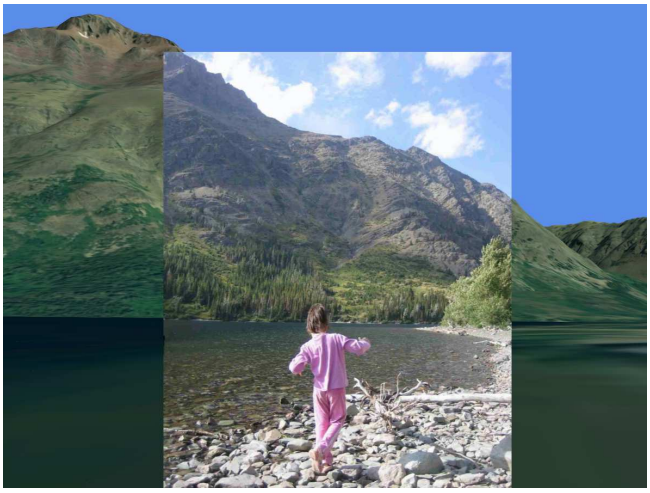


Figure 1: A frame from a 3D slideshow. Notice how the landscape smoothly transitions between the digital terrain and the photo.

While general location information is helpful, we believe that even more precise location information, namely the exact position and orientation of the camera, can give an even greater benefit to users. In this work, we achieve this goal by registering photographs to 3D digital models of the scenes from which they were taken. First, the digital models provide the anchors for precisely positioning images. Once registered, the models create a context for each photo, showing not only where it was taken but also its surrounding environment. This context enables more informative ways for photo story-telling and browsing. For example, instead of a typical slide-show experience in which photos are blended into each other, photos could be shown in situ in the 3D environment, with a virtual camera flying to each position and showing the appropriate photograph. Figure 1 shows one frame of this 3D

slide-show. Notice how an accurate registration preserves the continuity of the landscape as it transitions from the photo to the digital model.

In this work we explore the process of and tools for registering photos to geo-referenced digital terrain and building models. This work is only now becoming possible given both the prevalence of low cost GPS devices which enable the approximate positioning of photos as well as the recent availability of large amounts of geo-referenced models. Several companies, as well as the U.S. government, are now serving this data to the public. Despite the benefits of registering photos to this data, no consistent or standard solution has emerged.

In this paper, we present two interfaces for users to align loosely positioned images to precise geographic models; one which uses 2 separate views, one of the image and one of the model; and the other which uses one blended view in which the image is overlayed on top of the models. Figure 2a shows the overlay interface, in which an image is *overlayed* on top of a 3D view. The overlayed image can be made semi-transparent so that the user can see through to the 3D view. By "3D view" we mean a 2D rendering of the 3D models. Henceforth, we will use 3D view to denote this meaning. The motivation behind using an overlay is to provide quick feedback on the current alignment.

Figure 2b shows the split interface, in which an image is placed next to the 3D view. This a common interface used in the photogrammetry and vision community. The related work section discusses this in more detail.

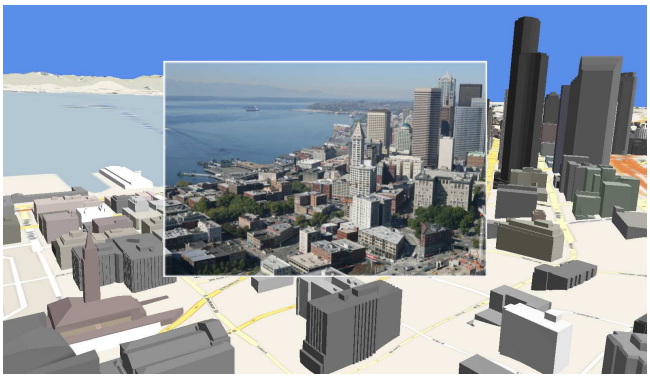
We conducted a quantitative comparison of these two interfaces and present the results of qualitative feedback as well. The results of our study suggest a redesigned interface that combines the best aspect of each interface based on our findings.

RELATED WORK

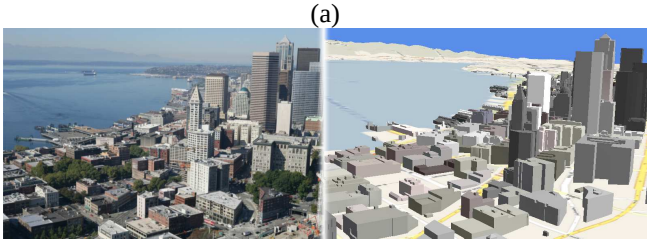
Techniques for registering an image to digital models can be split into two categories: automatic and manual techniques. Automatic techniques attempt to automatically find correspondences between an image and 3D models, but make strong assumptions about similar lighting and texture, or the number of necessary images. Manual techniques have been largely introduced in the context of accomplishing a particular task, like camera control, with little attention on the effectiveness of the interface. We briefly review automatic and manual techniques for registering images to 3D models.

Automatic techniques

One effective family of automatic techniques for registering images to 3D models is Structure from Motion (SfM). In SfM, the 3D models are assumed to be unknown, and solved in addition to the registration. More specifically, SfM seeks to find camera parameters and 3D models that are consistent across all images [8]. These camera parameters allow the appropriate images to be registered to the 3D models. Several works based on SfM [13, 3, 12] have robustly recovered camera calibration and 3D models from varying photos. Unfortunately, these algorithms still require a collection of images



(a)



(b)

Figure 2: In (a), an image is overlayed over the 3D view. In (b), an image is displayed next to the 3D view. For clarity, the models have been rendered without their textures.

with similar lighting and texture.

The interfaces presented in our work are effective for single images, relying on the user's expertise in recognizing corresponding features under varying lighting and texture. We are able, for example, to register an abstract painting to our realistically-rendered 3D models.

Manual techniques

Manual techniques for registering an image to 3D models rely on the user being in the loop. Such techniques are used in a variety of communities, including computer graphics, augmented reality and photogrammetry. However, very little attention has been given to the effectiveness of the interface in terms of accuracy and time to completion.

In computer graphics, registering an image to 3D models has been used for camera control, whereby users specify where 3D points in the world should project to in the image. Gleicher and Witkin introduced "through-the-lens" camera control [7], in which a user controls a virtual camera by specifying dragging motions in the image. Researchers have since modified this form of control for virtual directing of virtual objects [2, 10]. Online mapping services [4] also offer through-the-lens camera control for roughly placing a virtual camera in the world. However, they currently do not accurately register the image to the 3D view.

In augmented reality, video [15] or optical feeds [16, 5] are registered to 3D models to calibrate the head-mounted-display to the world. In this case, the "image" being registered is the video or optical feed.

In photogrammetry, satellite or orthographic imagery for maps

are registered to digital terrain for rectification [6, 14]. These tools commonly use the split interface. In our user study, we use an extended split interface that includes a mode in which the current registration can be overlayed on top of the 3D models, providing visual feedback.

The study presented in this paper compares the split and overlay interfaces and suggests improvements to these interfaces for the task of registering images to 3D models. Before discussing the interfaces, we describe the underlying camera model that is common to both. This camera model enables the system to update the 3D view so that it registers to the image.

CAMERA MODEL

An image is registered to a 3D model by specifying a set of at least 5 corresponding pairs of points between the image and the model. We defer the discussion of the different user interfaces for the matching to the next section. Both interfaces enable the user to match between the image and the 3D view.

The 3D view is created by rendering from a virtual camera. The virtual camera represents the initial guess for the parameters of the actual camera that took the image. As more correspondences are specified by the user, the virtual camera is updated. When the views are aligned, the parameters of the virtual camera will be the same as the parameters of the camera that took the photo.

The virtual camera uses a camera model, which projects points in the 3D world to points on the image plane. In our implementation, a camera is represented by a camera position, orientation, and focal length (skew angle is assumed to be zero, and the principle point is assumed to be in the center of the image):

$$\begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} [\ \mathbf{R} \quad \mathbf{t}\ ] \mathbf{X} = \mathbf{x} \qquad (1)$$

where f, and $\mathbf{R} \quad \mathbf{t}$ refer to the camera's focal length, rotation, and translation, respectively. $\mathbf{X}$, a point in the world, is projected by the camera model to $\mathbf{x}$, a pixel location.

By matching corresponding 2D points in the image ($\mathbf{x_i}$) and 3D points on the model ($\mathbf{X_i}$), the user specifies a set of equations. Camera parameters (focal length f, position $\mathbf{t}$ and rotation $\mathbf{R}$) are calculated by solving Equation 1. Equation 1 has 7 unknowns, and therefore needs at least 4 correspondences as constraints (2 equations for each correspondence).

The user selects 3D points on the model by clicking on 2D locations in the 3D view. Given the parameters of the current virtual camera, we can cast a ray through the camera center, through the pixel location and intersect it with the 3D model to recover the corresponding 3D point.

When the user begins registration, there are fewer constraints than unknowns. In these cases, the system solves for a subset of the camera parameters, fixing the other unknowns at reasonable values. We chose to solve the parameters in the following order: camera orientation, focal length, and position. Empirically, we found this ordering to be the most intuitive for users. We now detail how we solve for each parameter and describe its visual effect for the user. We will use the photograph in Figure 3 as an example. Figure 4 shows the starting position in the 3D view.

Figure 3: A photograph of the Seattle skyline, downloaded from the web.

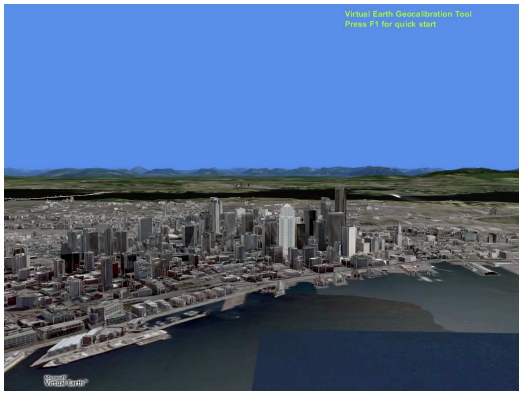


Figure 4: The initial 3D view, rendered using the virtual camera.

The first pair of corresponding points in the image and the 3D view affect the camera's pitch and yaw, assuming that the initial position and focal length are approximately correct. The resulting virtual camera creates a new 3D view of the models, rotated such that the first pair of points align on the same position in the view. Figure 5 illustrates this rotation. The correspondence is shown as a red pin the 3D view. The other corresponding position (in the photograph) is not shown.

After adding the second corresponding pair of points, the system solves for the focal length, as well as all 3 angles of rotation. Visually, the 3D view can rotate, while the models can scale, according to the distance between the two correspondences. For example, if the two correspondences are placed far apart, the models will scale up. If the two are placed close together, the models will scale down in the 3D view. Figure 6 shows how the models scale up for the second correspondence.

With the addition of the third corresponding pair, this enables the estimation of the camera position, using the previously estimated focal length and orientation. We use the minimal-

Figure 5: 3D view after corresponding 1 pair of points. Notice the camera has rotated so that the tip of the white building in the 3D view is at the same position in the photo.

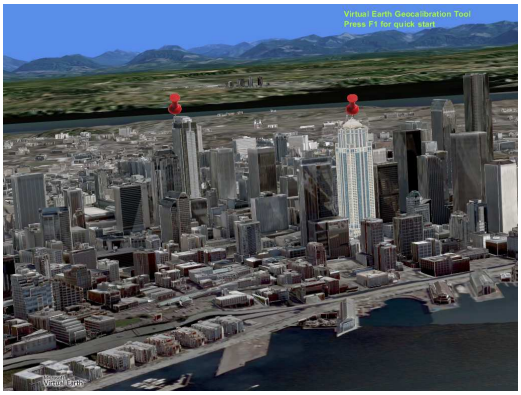


Figure 6: 3D view after corresponding 2 pairs of points. The models in the 3D view have scaled up to match the two pairs of correspondences. Notice that the first correspondence is still fixed in the same position.

Figure 7: 3D view after corresponding 3 pairs of points. The virtual camera has only changed in position to match the correspondences. The camera uses the previously estimated focal length and orientation.

Figure 8: 3D view after corresponding 4 pairs of points. The virtual camera adjusts its position, orientation and focal length to match the correspondences. For example, the shoreline has moved up in the 3D view. Notice that this 3D view is quite similar to the original photograph.

solver technique introduced by Nistér and Stéwenius [11]. Figure 7 shows the change in the 3D view.

When four or more correspondence pairs are available, there are enough constraints to solve for all the camera parameters. Camera calibration techniques such as [1] are used to solve for camera position, orientation, and focal length. Figure 8 shows the 3D view after 4 corresponding pairs are specified.

As the user specifies more than 4 correspondences, there will be more constraints than unknowns. In this case, the system attempts to solve all constraints simultaneously, spreading the error over all correspondences. Since not all constraints can be satisfied, sometimes the user will observe a small shift in the correspondences he specifies. However, in most cases, if the user is close to the correct camera model this shift is small and negligible.

The two interfaces we designed, overlay and split, enable the user to specify the corresponding points between the image and the 3D view. We describe these next.

USER INTERFACES

We now present two interfaces that enable a user to register an image to digital models. In both interfaces, we assume that the user has already found an approximate position, so that corresponding features can be found in both the image and the 3D view. This approximate position can be obtained by clicking on a 2D map, by specifying a latitude and longitude, or from a GPS device.

Overlay interface

In the overlay interface, we place the image over a 3D view of the digital models. The image can be thought of as lying on a foreground layer and remains static. The 3D view lies on the background layer and can change, depending on its camera parameters. To allow the user to see through the image onto the 3D view, the image is rendered with an alpha, which specifies the level of blending between foreground and background. An alpha of 0 turns the foreground image transparent, revealing the background 3D view. An alpha of 1 makes the foreground image fully opaque. The user can “flicker between the layers” by interactively adjusting the opacity of

the image. This is performed by quickly tapping a key, which toggles the image's opacity between 0 and 1. Intuitively, one can think of the image as a slide transparency, taped onto the front of the screen. Displayed in the screen itself is the 3D view.

Figure 9 shows an overview of the registration process using this overlay view. In 9a the image starts as a thumbnail over the 3D rendering. As mentioned earlier, the user starts in an approximate position. The user enters the overlay interface by pressing a key to place the image on top of the 3D view. This is shown in 9b. Her first task is to identify misregistration and she flickers between the layers to do so. Misregistration will pop out as motion as she flickers the image.

To correct a misregistration, in the 3D view she drags the point towards its location in the image. To help the user find the target in the image, the image automatically becomes more opaque when she drags. Figure 9c shows the user's mouse and her dragging motion.

When the user has aligned the points in the 3D view and the image, she releases her drag and places a pin at that location, by right clicking. This is shown in Figure 9d. Figure 9e shows that the pin locks the same position in the image. The pin tells the user that this point in the image is fixed and will not move. For the system, this pin becomes a permanent 2D-3D constraint that is an input to the camera calibration.

As the user pins more correspondences, the virtual camera is updated such that the 3D view matches all correspondences as best as possible. This process continues until the user is satisfied with the overlap between the image and the 3D view. Figure 9f shows the final result.

Split interface

Another approach to registration is to find corresponding points between the image and the 3D view when both views are shown side by side. Figure 10 shows an overview of the technique.

In Figure 10a, the user starts with a split view, the image on the left, the 3D view on the right. She begins by visually finding correspondences between both sides. After identifying a correspondence, she places a pin on both sides, as shown in Figure 10b. The system automatically updates the 3D view so that the pins are in the same position in both views. The user continues to add correspondences, as shown in Figure 10c. To check the accuracy of the registration, the user can drag the 3D view on the right onto the image on the left, producing an overlay view. She can toggle the opacity, like in the overlay interface, to detect misregistration. If there is misregistration, she can split the views and continue to add corresponding points. Figure 10c shows the result of corresponding 5 points. The final overlay showing the correct registration is shown in Figure 10d.

USER STUDY

Because each of the aforementioned interfaces have different features, each have their own relative advantages and disadvantages. The overlay interface provides a large display surface where it renders an integrated representation of both an image and a 3D view. In contrast, the split interface offers smaller separate views for both the image and the 3D view. Also the overlay interface requires users to engage in dragging tasks to align corresponding points, while the split interface only needs users to click on matching pairs of features. To learn more about these different registration interfaces, we conducted a user study that seeks to measure users' performances during registration tasks using both the overlay and split interfaces. We are interested in observing how long an average registration task might take and what level of accuracy a user can achieve when using one of these interfaces. At the same time we hope to gather intuition about the interfaces from the users' qualitative impressions of each one.

Apparatus and Participants

We used two systems running Windows XP and Vista, with 3.2 GHz processors, 3.0 GB of RAM and a 1600 $\times$ 1200 LCD display with an optical mouse. Three females and thirteen males, 19-50 years old, participated in the study and were recruited though e-mail solicitation within our company. All participants, save one, work with computers on a daily basis for over 8 hours. Participants were given a $7 meal certificate as compensation.

Task and Stimuli

Each task consisted of a participant registering a photo using either the overlay or the split interface. At the beginning of each task, participants start with the photo and a 3D view that loosely matches the photo. The initial, approximate virtual camera for the 3D view is determined by offsetting the camera by 10 meters, tilting or turning the camera by up to 15 degrees, and initializing its field of view to 45 degrees. These approximations are common noisy estimates from GPS devices and for manual geotagging of image orientation. Participants then register the given photo to the given 3D models by specifying multiple points of correspondence between the photo and the 3D view. The way in which participants specify these correspondence points depends on the particular interface as we explained in the previous sections.

Each photo used in the study belongs to a set of 8, which we divided in two. Photos are taken from the sky, to reduce user confusion if the virtual camera is accidentally put underground or inside buildings. Figure 11 shows the two sets of photos.

Procedure and Design

We used a 2 *Technique (Overlay, Split)* within-participants design. Each user was presented with four registration tasks per technique, using the same set of four photos. The dependent variables were *Interaction Time* and *Error*. We computed the interaction time as the time between the start of a participant's interaction and the moment when the participant clicks a "Next trial" button.

We computed the error as an image-based reprojection error. Previous to the experiment, an expert user registered by hand all images to subpixel accuracy. This registration provides the ground truth parameters of the camera model for each image. We can generate a ground truth 3D view using the ground truth camera, for each image. Ideally, the error should get closer to zero as a user's 3D view aligns to the

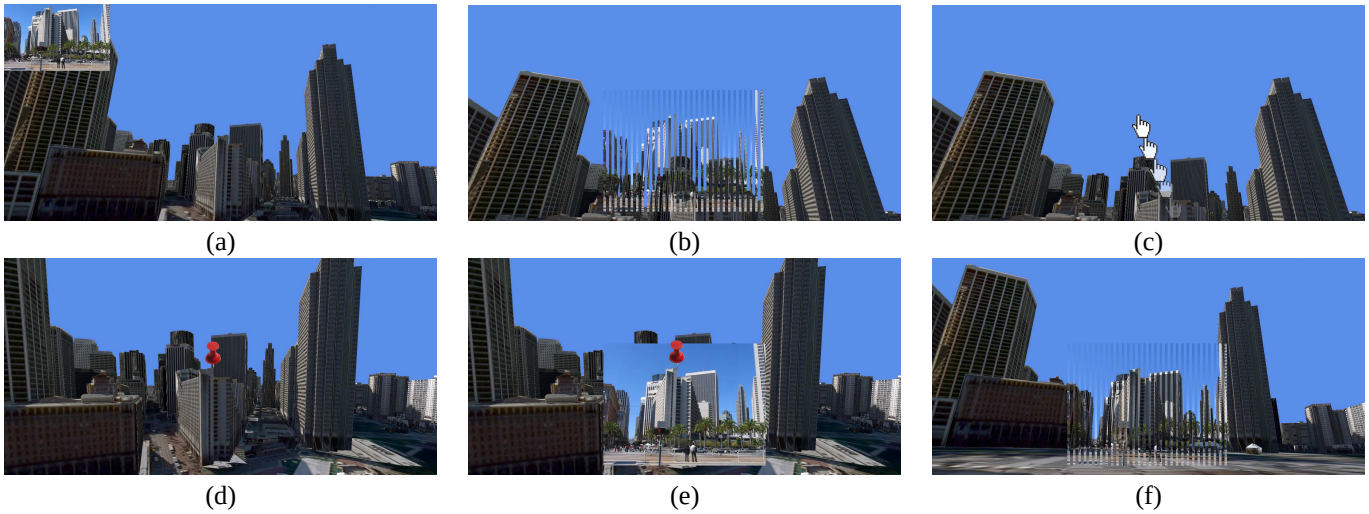


Figure 9: Overlay interface. (a) shows the initial thumbnail of the image. (b) shows the image overlayed on top of the 3D view. For clarity, instead of blending, the figure shows alternating strips of the image and the 3D view. In the application the user adjusts the opacity of the image. (c) shows a mouse drag to correspond two features. (d-e) show the pin locking the feature in both the image and the 3D view. (f) shows the final result after adding four more pins (pins are not shown).

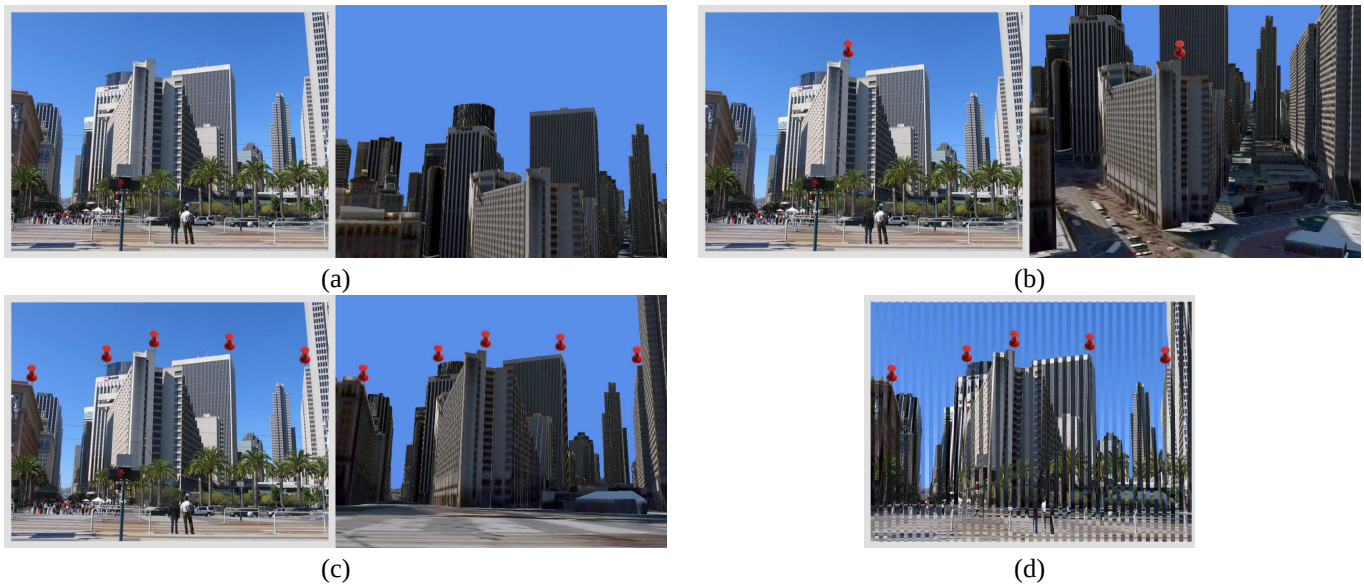


Figure 10: Split interface. (a) shows the starting position of the interface. The photo is shown on the left, and the 3D view on the right. When the user clicks two corresponding points, one on the left, the other one on the right, the system updates the 3D view on the right. Now the position under the pin is at the same pixel location in both views, as shown in (b). (c) shows the result after corresponding 5 points. The resulting registration is shown in (d).

ground truth 3D view. To measure this error, we first take a set of 100 uniformly sampled locations in the user's 3D view, project them onto the 3D models, then onto the ground truth 3D view. We measure the average L2 distance between the user's projected points and the ground truth points in the 3D view. This average is the error reported during the registration task.

In addition to these measurements we also kept a log of all the input events and the times at which they occurred. In summary the experiment consisted of:

16 participants $\times$ 2 techniques $\times$ 4 images = 128 registration tasks.

Trials were divided into two blocks, one per technique. Before the four trials for a particular technique, we instructed participants on the functionality of a particular interface and gave them time to perform two practice registration tasks. The presentation order for techniques was counter-balanced across participants.

Results

The study took an average of one hour per user. We performed a two-way repeated measures analysis of variance (RM-ANOVA) on Interaction Time and Error. We found no

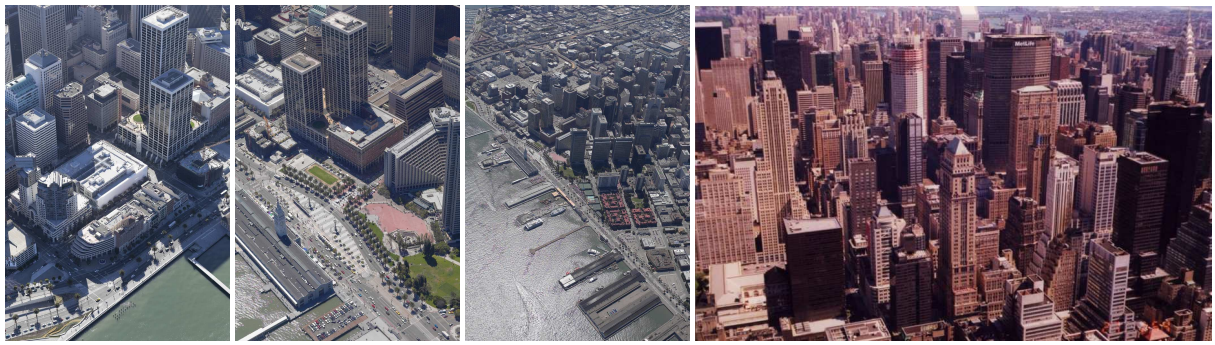


Image set A

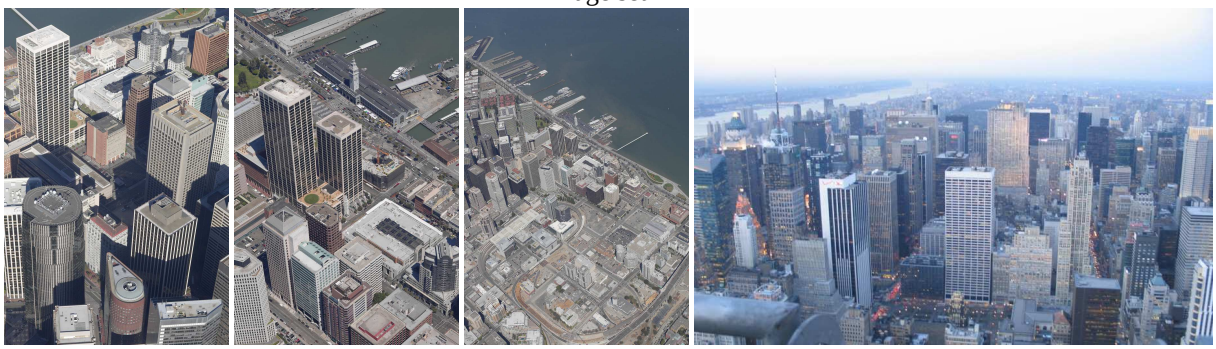


Image set B

Figure 11: Image sets used for the user study.

significant effect for technique on Interaction Time ($F(1,10)=1.82$; $p=0.20$). We did observe a significant effect for Technique on Error ($F(1,10)=12.83$; $p=0.005$) with the split interface yielding final error magnitudes of 4.121. Figures 12 and 13 illustrate these results.

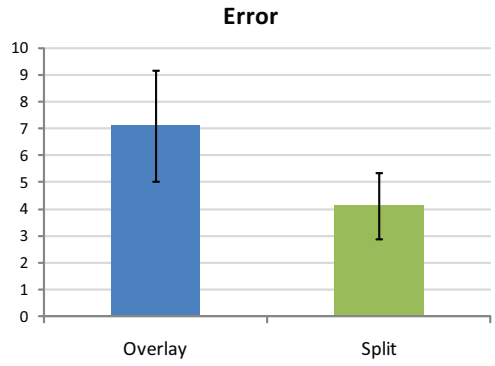


Figure 12: Average error distance vs. interface with 95% Confidence Intervals (CI)

At the end of the trials for each interface we solicited qualitative feedback from the participants. We asked participants to grade their experiences using a Likert scale ranging from 1 (strongly disagree) to 5 (strongly agree). The participant's scores regarding all aspects of their experience reflect some of the results from our quantitative analysis: all scores are not significantly different for each technique. Figure 14 illustrates.

The similarities in scores between the two interfaces res-

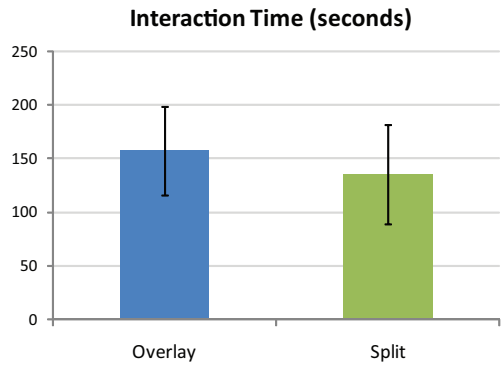


Figure 13: Average trial time vs. interface with 95%CI

onates with the participants' comments, which are balanced in terms of preferring one interface over the other. For example, while a participant commented "I did not enjoy the blended view. Manipulating a 3D scene is so much more difficult than clicking points.", another said "In the overlay view, you get realtime feedback, you see the 3D image rotating and zooming to fit the photograph, and you miss that in the split view". In general many appreciated the real-time feedback that the overlay view provides, while at the same time expressed some dissatisfaction when occlusion got in the way of clarity. A participant summarized this by saying "It is nice not having them on top each other [in the split view]. Like before [in the overlay view], the Empire State building blocked my view". Other users preferred the overlay when the registration was close for images with many repeated structures, like buildings. In the split interface, the

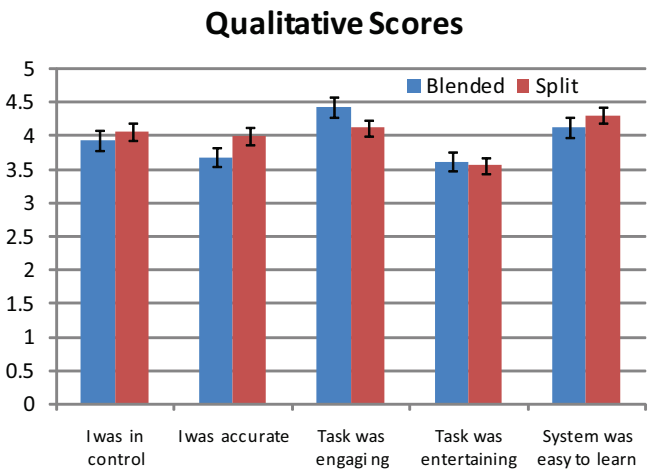


Figure 14: Average qualitative scores +/- 1 Stdv

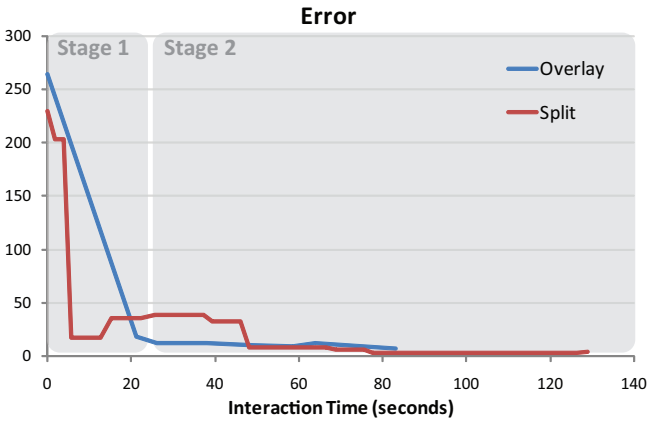


Figure 15: Error over time for a particular photo and user. We highlight two regions: a rapid-convergence region (stage 1) and a refinement region (stage 2).

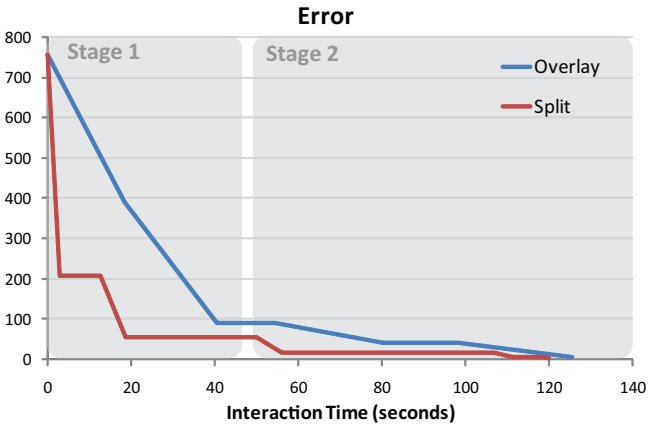


Figure 16: Error over time for another photo and user. We highlight two regions: a rapid-convergence region (stage 1) and a refinement region (stage 2).

repeated structures made it difficult to correspond points. Another issue was that participants were disoriented when the 3D view changed abruptly in the split interface, "When adding a pin, the right hand side jumps, which is very disconcerting".

Along with these contrasting points of view, many participants found the techniques as complementary and helpful at different stages of the interaction "With [overlay], I feel I can use fewer points. I get more feedback. If I could initially add a few points [in the split view], then go into [overlay view] to adjust it, that would be great" or "It would be great to initially align [the images] with a split, perhaps with 2 [pins], for scale, so that it is not so initially wrong". This feeling could explain why people gave both techniques similar scores and suggest to us that it is worthwhile to explore the design of an hybrid registration solution where users first use a split interface to achieve a coarse global correspondence and then progress to a blended view for subsequent fine adjustments. Figures 15 and 16 show the errors over time of two representative cases of registration for two different photos, by two different participants. While these images do not provide hard quantitative evidence, they still suggest that there are two distinctive stages during the registration process: an initial, rapid-convergence stage; and a later, refinement stage. These images also indicate that it is not unreasonable to think of an interaction where the interface switches from split to overlay after a particular error threshold is crossed.

In addition to specific issues with each particular interface, participants almost unanimously identified missing functionality that they believe can improve the calibrating experience. Chief among these features is the ability to be more precise at pin-pushing, by either being able to zoom into the photo or model, or by providing an intelligent snapping to distinctive features in an image, such as an edge or corner.

In summary, although the overlay interface was less accurate, qualitative data suggests that it provided a level of engagement that resonated well with participants. For applications like casual photo annotation, this quality may be useful, since in such scenarios high accuracy is not crucial. Furthermore, in scenarios where many photos may need to be annotated, an interface that elicits a high level of engagement can be beneficial. Results from this experiment also suggest the utility of a hybrid interface, combining both the overlay and split interface. The split interface could help to quickly obtain coarse registration. The overlay interface could be useful in refining the registration in a later stage.

APPLICATIONS

In this section we showcase a variety of applications that make use of the registration techniques described earlier. Either interface can be used to perform the registration.

Slides shows with context: The model itself instantly provides context to the photographs. As seen in Figure 1, one can use the models to help tell a story by transitioning from one image to the next using the 3D as the backdrop for the photographs.

Exploring varying lighting conditions: Registering a photograph taken under lighting conditions that are different than the digital models does not present additional problems since the human is in the alignment loop. Figure 17 shows a pho-

tograph taken at dusk. The surrounding digital models use textures extracted from photos taken at varying times during the day. Using either a split or overlay interface, users can still register such photos to the digital models.

Figure 17: Registering a photograph taken at dusk. Note the lighting in the photograph is different than in the 3D models. However, we are still able to successful register the photograph.

Placing archival images in a modern context: In older images, not only is the illumination different, but many buildings are missing in the photograph. Figure 18 shows an archive photograph of a construction worker atop the Empire State building. At the time this photo was taken, many buildings were not built in the current digital model dataset. However, the user was still able to successfully register the photograph using features from buildings that still exist today.

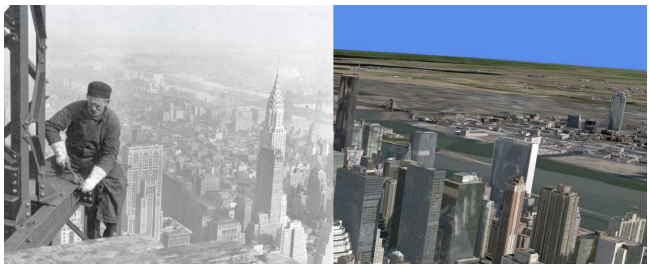


Figure 18: An archive photograph illustrating the construction of the Empire State building. Photos such as these can be registered to the modern digital models.

Positioning a painting in the world: Finally, the example in Figure 19 shows a painting registered to digital terrain. The painting has varying lighting and geometry due to the painter's interpretation of the scene. In addition, it only approximates a perspective projection. However, a user can still register the painting convincingly into the digital terrain.

Video registration: Static video can also be registered to the digital terrain and building models. In Figure 20, a live traffic-camera feed is registered to its appropriate location in the digital terrain. Accurate registration gives a context to the traffic; we better understand how the traffic enters and leaves the video feed. With multiple video feeds we may even be able to visually extrapolate the traffic to unobserved areas.

Figure 19: A painting registered to the digital models. We can register paintings that have have an approximate single center of projection. After registration, we have a feeling of the artist's point of view when he created his painting.

Figure 20: A frame from a live traffic camera feed. Accurate registration lets us understand how the traffic in the video enters and leaves the surrounding context.

Aligning sets of overlapping images: In previous examples, photos are registered individually to digital models. If these photos belong to a collection, they may be calibrated relative to each other using structure-from-motion (SfM) techniques [13]. In this case, the user interactively registers 4 or more photos and the rest of the collection of photos may be automatically registered to the digital models. The first 4 photos are used to compute the rotation, translation and scale that map from the SfM coordinate system to the digital models coordinate system [9].

CONCLUSIONS AND FUTURE WORK

We present interactive techniques for accurately registering images to georeferenced 3D digital terrain and building models. We enable the user to perform the registration through two novel interfaces, which we call overlay and split. Once registered, we have shown a few applications the aligned images can afford.

A user study comparing the two registration interfaces showed that the split interface was more accurate. However many users enjoyed the real-time feedback from the overlay interface, when the views were not cluttered. Many participants found the interfaces complementary, suggesting a hybrid one.

Figure 21: Semi-automatic photo registration. 4 images are manually registered to the digital models. The remaining 16 images are automatically registered to the first 4 and each other using structure-from-motion [13]. The automatically registered images are shown.

Based on these findings we are seeking ways to improve each interface and are considering merging both. For example, in the overlay interface, we are investigating ways to reduce the clutter by using non-linear blending techniques that preserve semantically important areas. When dragging the mouse to correspond a feature in the image, the interface currently fades out the digital models. Instead, the interface could keep semantic information (such as the building the user is dragging), and fade the less locally salient information.

We also are developing a hybrid interface, whereby the user initially performs coarse alignment via the split view, then refines the result using the blended view. An initial study indicates improved time to convergence at the same accuracy. Both interfaces could make use of zooming and automatic snapping. When a user picks a point in the image, we snap that point to local corners or edges. This would help the user obtain sub-pixel selection.

We have only touched on the kinds of applications that can arise given registered images. We are also investigating how to adapt these registration techniques to handle video, especially video from moving cameras. The challenge is to enable accurate registration without having the user register each frame.

One can expect that as interfaces such as the ones shown here become easy enough to use that millions of pieces of visual media will be registered to the ever growing 3D representation of the world. Many exciting opportunities and challenges will arise in developing ways to browse and tour through these media both for entertainment and information gathering.

REFERENCES

1. Y.I. Abdel-Aziz and H.M. Karara. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. *Symposium on Close-Range Photogrammetry*, 1971.

2. William Bares, Scott McDermott, Christina Boudreaux, and Somying Thainimit. Virtual 3d camera composition from frame constraints. *ACM Transactions on Multimedia*, 2000.

3. Matthew Brown and David Lowe. Unsupervised 3d object recognition in unordered datasets. *International Conference on 3D Digital Imaging and Modeling*, 2005.

4. Google Earth. 2008. earth.google.com.

5. Yakup Genc, Frank Sauer, Fabian Wenzel, Mihran Tuceryan, and Nassir Navab. Optical see-through hmd calibration: A stereo method validated with a video see-through system. *IEEE and ACM International Symposium on Augmented Reality*, 2000.

6. PCI Geomatics. 2008. www.pcigeomatics.com.

7. Michael Gleicher and Andrew Witkin. Through-the-lens camera control. *SIGGRAPH*, 1992.

8. Richard Hartley and Andrew Zisserman. *Multiple View Geometry*. Cambridge University Press, 2004.

9. Berthold K. P. Horn, Hugh M. Hilden, and Shahriar Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. *Optical Society of America*, 1988.

10. Eric Marchand and Nicolas Courty. Image-based virtual camera motion strategies. *Graphics Interface*, 2000.

11. David Nistér and Henrik Stewénius. A minimal solution to the generalized 3-point pose problem. *Journal of Mathematical Imaging and Vision*, 2006.

12. Frederik Schaffalitzky and Andrew Zisserman. Multi-view matching for unordered image sets, or ”how do i organize my holiday snaps?”. *European Conference on Computer Vision*, 2002.

13. Noah Snavely, Steven M. Seitz, and Richard Szeliski. Photo tourism: Exploring photo collections in 3d. *SIGGRAPH*, 2006.

14. ITT Visual Information Solutions. 2008. www.ittvis.com/envi.

15. Mihran Tuceryan, Douglas S. Greer, Ross T. Whitaker, David E. Breen, Chris Crampton, Eric Rose, and Klaus H. Ahlers. Calibration requirements and procedures for a monitor-based augmented reality system. *IEEE Transactions on Visualization and Computer Graphics*, 1995.

16. Mihran Tuceryan and Nassir Navab. Single point active alignment method (spaam) for optical see-through hmd calibration for ar. *IEEE and ACM International Symposium on Augmented Reality*, 2000.