ConstructAide: Analyzing and Visualizing Construction Sites through Photographs and Building Models

Supplemental document

Kevin Karsch Mani Golparvar-Fard David Forsyth
University of Illinois

Figure 1: *ConstructAide results on the Basement dataset, demonstrating architectural rendering, performance monitoring, and temporal navigation.*

In this document, we describe the details of our constrained bundle adjustment procedure (used within our Model-assisted Structure from Motion technique). We also include further figures for visualization.

Appendix A: Constrained Bundle Adjustment

Bundle adjustment solutions for camera and geometry that differ only by a change of coordinate system (a gauge transformation) must have the same reprojection error. This effect is an important difficulty for systems that must produce general reconstructions. The effect is particularly pronounced as the percentage of camera pairs that view the same geometry goes down. In some cases even the structure of the gauge group is not clear [Snavely et al. 2008], and complex strategies apply [Bartoli 2003; Triggs et al. 2000]. Our case is simpler: we expect a high percentage of camera pairs to share features, and so we can resolve this issue by fixing the coordinates of one camera, the anchor camera [Hartley et al. 1992].

In typical SfM bundle adjustment formulations, reprojection error is minimized by simultaneously adjusting intrinsic and extrinsic camera parameters, and triangulated points X. Let $\mathbb{P} = \{\mathbb{P}_1, \ldots, \mathbb{P}_N\}$ be the set of all camera parameters corresponding to the N images, and tracks_i be the pixel locations of keypoint tracks in image i. The classic bundle adjustment problem is formulated as a nonlinear least squares problem,

$$\operatorname*{argmin}_{\mathbb{P},\, X} \sum_{i=1}^{N} \sum_{u \in \text{tracks}_i} ||\text{project}(\mathbb{P}_i, X_u) - u||, \tag{1}$$

where X_u is a triangulated point corresponding to pixel u, and $\text{project}(\cdot)$ is the function that projects 3D locations into 2D according to a set camera parameters.

We formulate a new version of this problem, *constrained* bundle adjustment, which leverages one or more calibrated cameras. In our system, the user provides guidance for registering a mesh to the initial image input to our system, giving a very good camera pose estimate for this image. We call this camera an *anchor*, and denote its parameters $\mathbb{P}_\dagger$. During bundle adjustment, this anchor camera is used to constrain the 3D points such that any point triangulated using a feature point from the anchor camera must lie along the ray generated by the anchor camera. Therefore, we re-parameterize points as $X_u(t_u) = \mathbb{P}_\dagger^{\text{center}} + t_u\, \mathbb{P}_\dagger^{\text{ray}}(u)$, where t_u is a scalar and $\mathbb{P}_\dagger^{\text{center}}$ is the anchor camera center and $\mathbb{P}_\dagger^{\text{ray}}(u)$ is the ray generated from pixel u in the anchor camera. Our formulation then becomes

$$\operatorname*{argmin}_{\mathbb{P}\backslash\mathbb{P}_\dagger,\, t} \sum_{i=1}^{N} \Bigg[\sum_{u \in \text{tracks}_\dagger} ||\text{project}(\mathbb{P}_i, X_u(t_u)) - u|| + \sum_{u \in \text{tracks}_i \backslash \text{tracks}_\dagger} ||\text{project}(\mathbb{P}_i, X_u) - u|| \Bigg]. \tag{2}$$

Notice that the anchor's camera parameters are left out of the bundle adjustment, and any tracks that are not seen by the anchor camera revert to the classic bundle adjustment formulation.

From our experience, this formulation typically provides better estimates since the model is constrained by accurate camera parameters. Also, it has an added benefit of having fewer parameters to optimize over, increasing optimization efficiency and reducing variance in the estimates. One downside is that this model can be inflexible if the other initial camera estimates are too poor, and we also propose a "soft-constrained" bundle adjustment in these cases:

$$\operatorname*{argmin}_{\mathbb{P}\backslash\mathbb{P}_\dagger,\, X} \sum_{i=1}^{N} \sum_{u \in \text{tracks}_i} w_i\, ||\text{project}(\mathbb{P}_i, X_u) - u||, \tag{3}$$

where w_i is a scalar weight dependent on each image. We set the anchor image's weight to a large value (e.g. 100), and all other image weights to 1, enforcing the reprojection error for the anchor camera to be much smaller than other cameras. This has a similar effect as Eq. 2, but allows for flexibility in the 3D point locations.

The user may also guide the registration of other images if certain conditions are met (see the main paper for details), adding new anchor cameras.

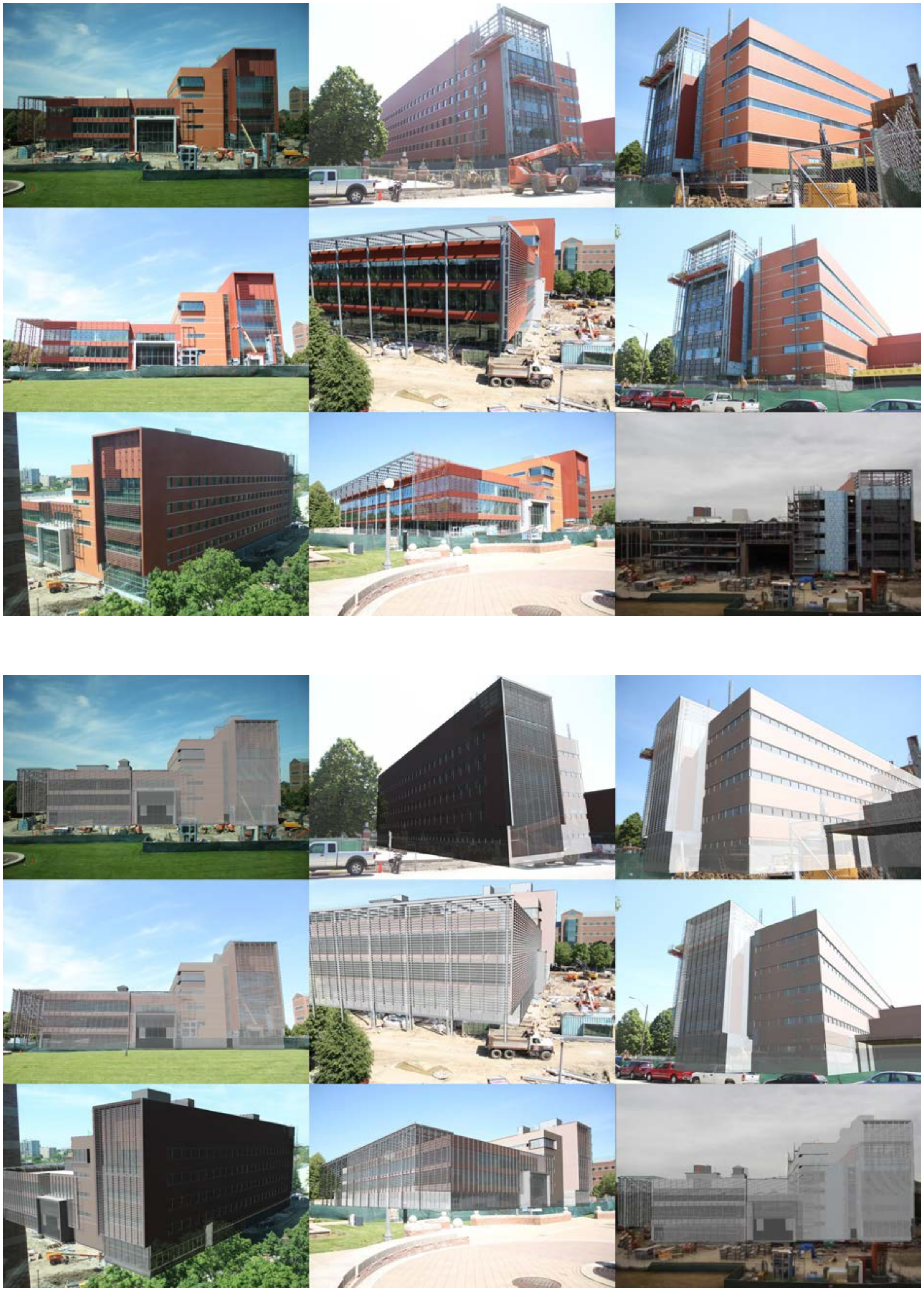

Figure 2: *Example registrations estimated with our Model-assisted SfM procedure. Original photos on top, rendered overlay demonstrating registrations on bottom.*

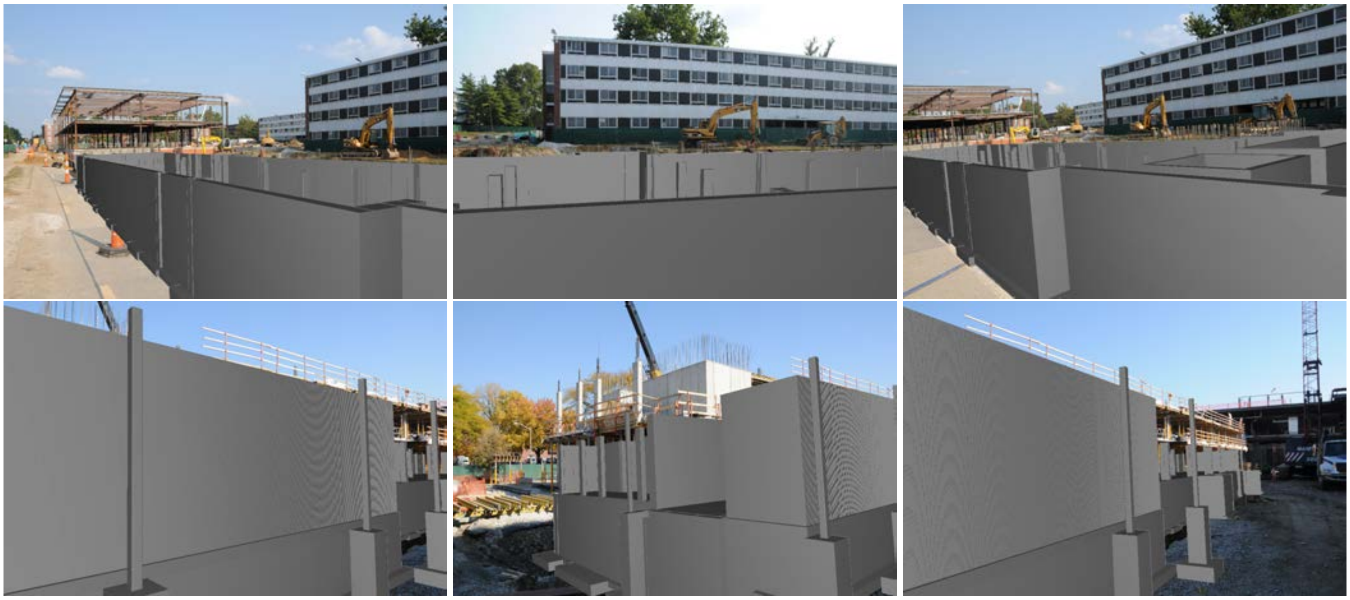

Figure 3: *Registration results from the RH4 A and RH10 datasets (used in our quantitative evaluation.*

Figure 4: *Additional results for other construction datasets – triangles indicate the rough camera pose of the data with respect to an overhead view of the construction sites, and border colors relate the cameras to surrounding photos. Our interface can help in generating job site summaries such as these. In the future, ConstructAide aims to automatically produce such summaries.*

References

BARTOLI, A. 2003. Towards gauge invariant bundle adjustment: a solution based on gauge dependent damping. In *Int. Conf. on Computer Vision*, 760–765.

HARTLEY, R., GUPTA, R., AND CHANG, T. 1992. Stereo from uncalibrated cameras. In *CVPR*, 761–764.

SNAVELY, N., SEITZ, S. M., AND SZELISKI, R. 2008. Modeling the world from Internet photo collections. *IJCV 80*, 2, 189–210.

TRIGGS, B., MCLAUCHLAN, P., HARTLEY, R., AND FITZGIBBON, A. 2000. Bundle adjustment a modern synthesis. In *Vision Algorithms: Theory and Practice*, B. Triggs, A. Zisserman, and R. Szeliski, Eds., vol. 1883 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 298–372.