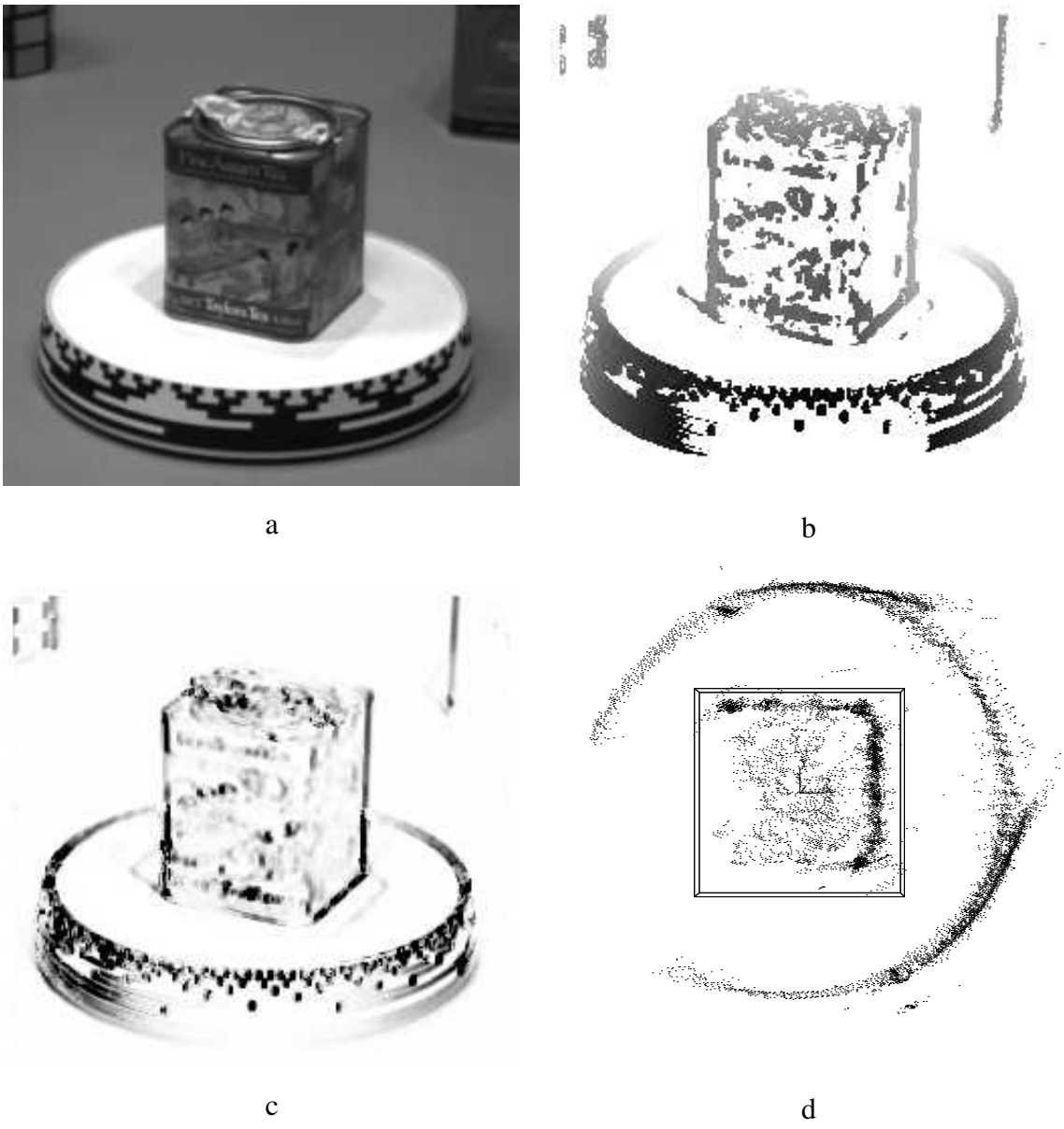
the domino cube (Figure 11) are roughly recovered. The structure of the dodecahedron (Figure 10) is more difficult to see from the top view.

The next step in the shape from rotation algorithm consists of merging neighboring 3-D points acquired from different viewpoints. Figures 12, 13, and 14 show the results of this merging step, operating incrementally on the complete 250 image sequences. We present this data as isolated points shown in 4 different projections: top, front, side, and oblique. Unfortunately, it is somewhat difficult to gauge the true shape of the object using these flat two-dimensional projections. In our own experiments, we can rotate the object interactively to get a good sense of the depth and relationship of the points (kinetic depth effect). We also use multiple colors to display points from different iterations (when examining the merging step).

From these figures, we can see that the overall shape of the objects is recovered well, although the exact surface data is not very smooth. Adding a small amount of image-plane smoothing should help to reduce this effect [Matthies *et al.*, 1989]. Of course, once a complete surface model is fit to this sparse data, the resulting solution will also be smooth. Figures 13b–d show that in some cases, shadows will be incorporated into the object model (under many imaging and lighting geometries, the shadows form non-rigid structures, and are therefore discarded by the temporal integration stage). To remove these shadow points, we could either cut off the bottom of the model, or use a more sophisticated color-based image preprocessing stage. More detailed examination of the data reveals that even after integration over many frames, the depth estimates near horizontal edges are still poor. This suggests a bias in our flow estimator, which could be reduced either by using more temporal averaging (antialiasing), using using spatio-temporal filters [Fleet and Jepson, 1989], or using variable-sized windows [Okutomi and Kanade, 1990].

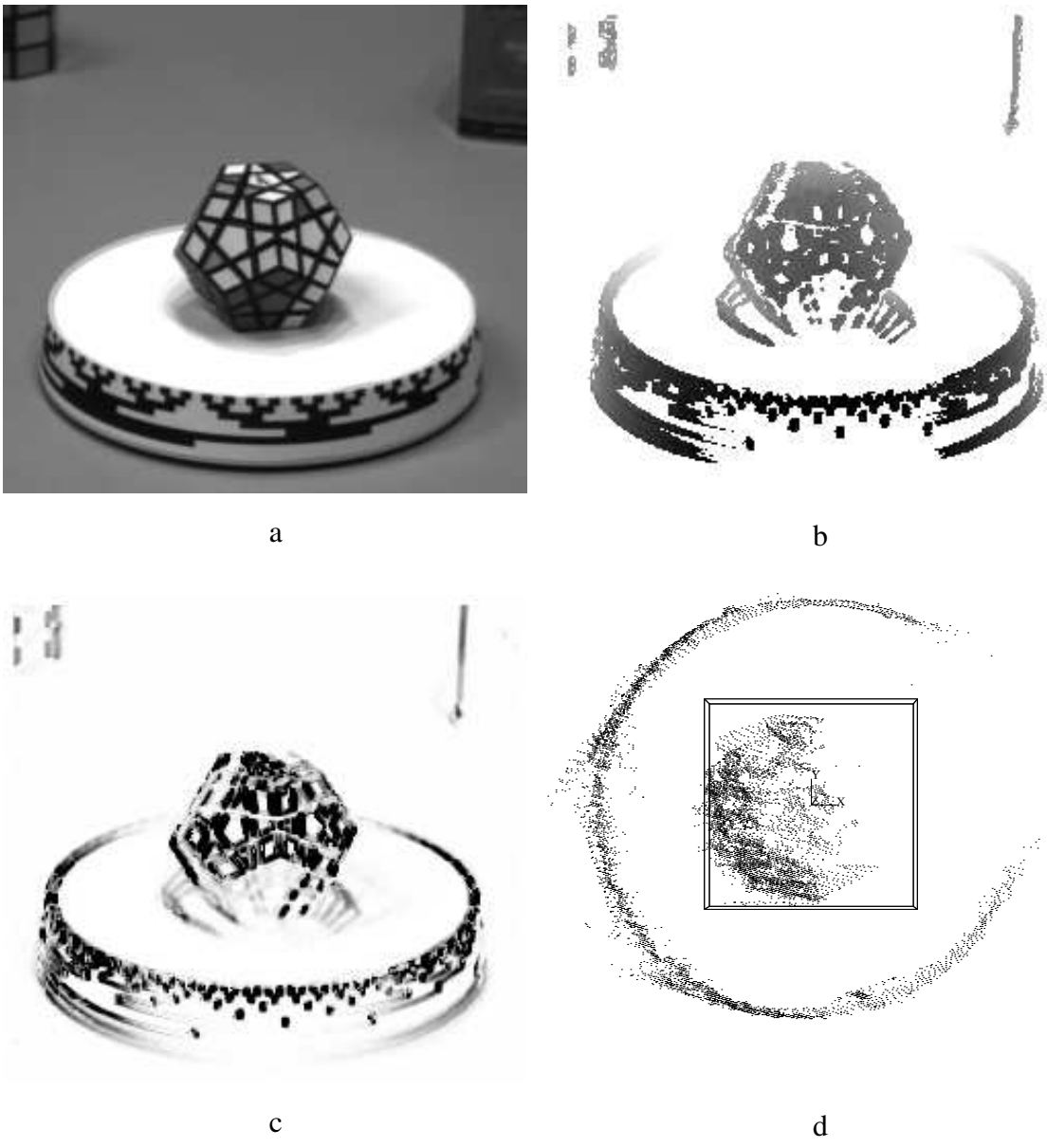


a b

c d

Figure 9: Flow computed from `assam` image series

(a) first image in sequence (b) depth map from flow (darker is nearer) (c) certainty in depth estimates $\sigma_z^{-2}$ (darker is higher certainty) (d) top view of 3-D point cloud

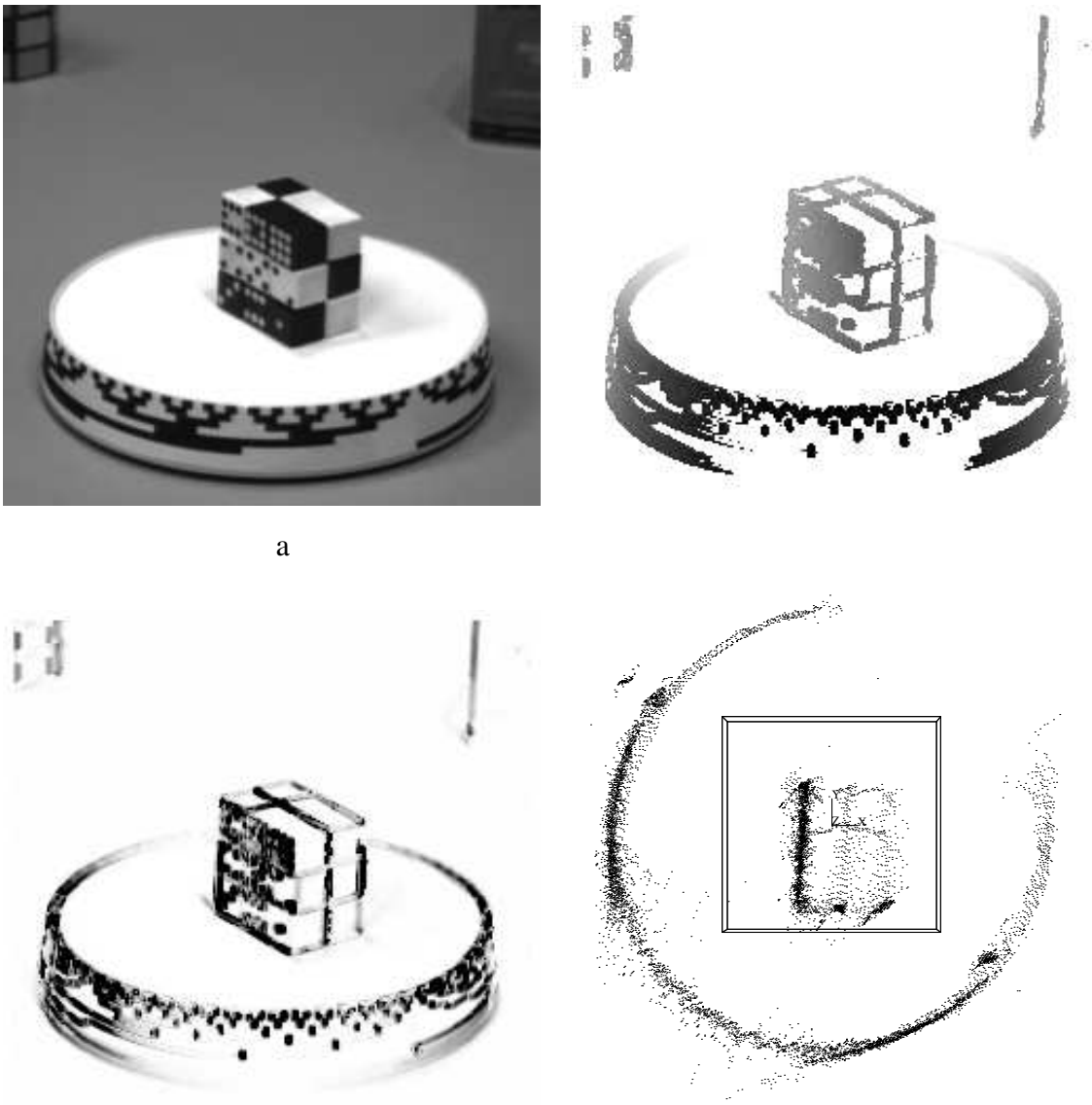


a b

c d

Figure 10: Flow computed from `dodecahedron` image series

(a) first image in sequence (b) depth map from flow (darker is nearer) (c) certainty in depth estimates $\sigma_z^{-2}$ (darker is higher certainty) (d) top view of 3-D point cloud



a c d

Figure 11: Flow computed from `domino` image series

(a) first image in sequence (b) depth map from flow (darker is nearer) (c) certainty in depth estimates $\sigma_z^{-2}$ (darker is higher certainty) (d) top view of 3-D point cloud

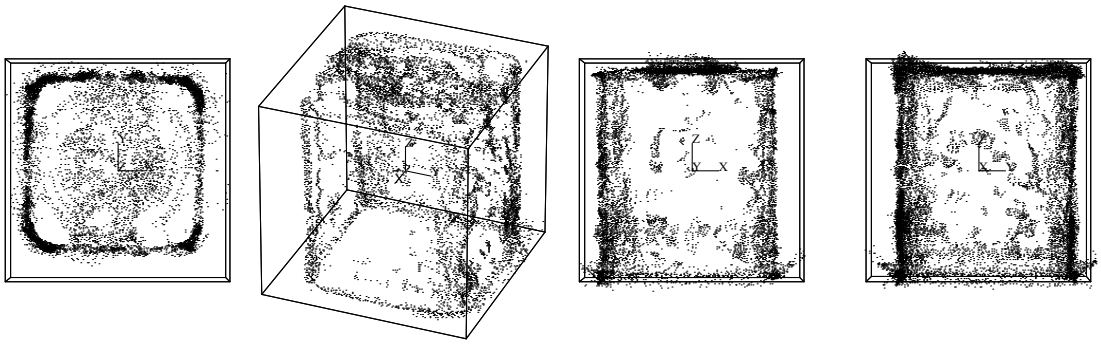



Figure 12: Final merged data from `assam` image series: (a) top view (b) oblique view (c) front view (d) side view

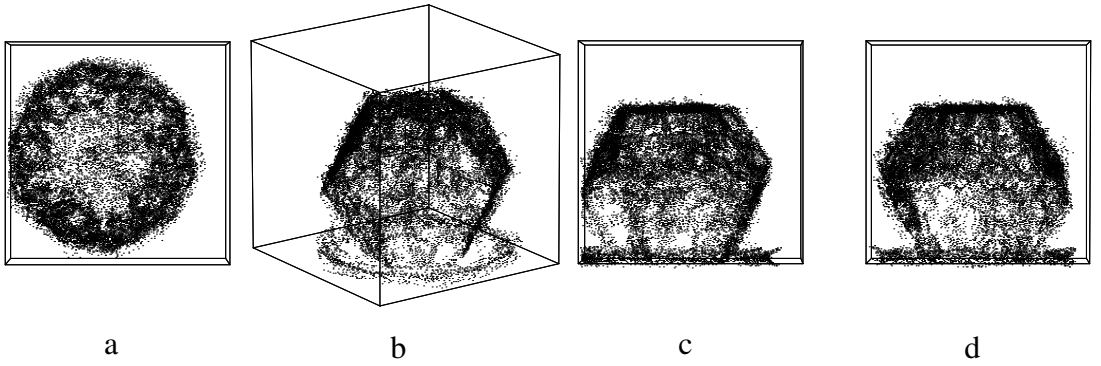



Figure 13: Final merged data from `dodecahedron` image series: (a) top view (b) oblique view (c) front view (d) side view

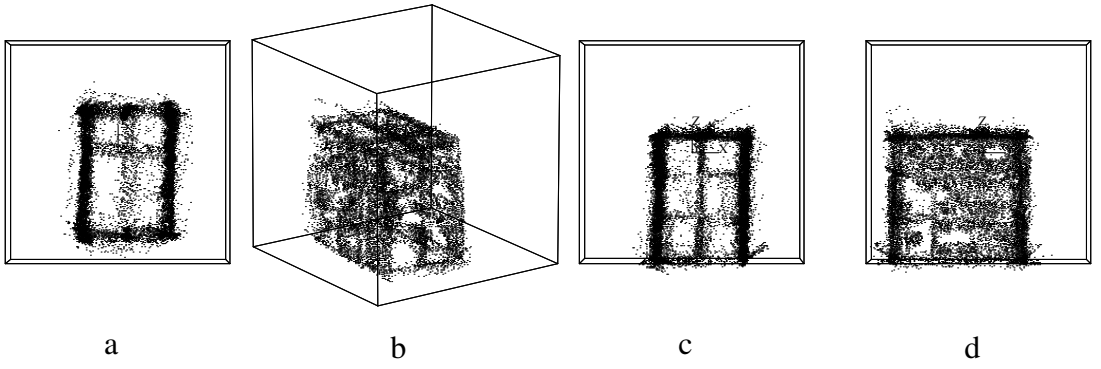



Figure 14: Final merged data from `domino` image series: (a) top view (b) oblique view (c) front view (d) side view

# 7 Discussion

The techniques we have described in this paper perform a shape construction task similar to that usually associated with active range sensors [Agin and Binford, 1976; Woodham, 1981; Besl and Jain, 1985]. An example of such a sensor, which can be used in short-distance indoor environments, is structured light, where an encoded light pattern falling on the object is used to give direct (and usually sparse) measurements of depth [Agin and Binford, 1976; Vuylsteke and Oosterlinck, 1990]. Compared to active range sensors, our approach requires a far less structured environment, since no special lighting sources are required, and the calibration of the system is simple and fairly automatic. Our technique also has the potential for better accuracy since our measurements are dense (at least in textured areas), and because we see more views of the object. On the other hand, our flow-based approach will fail in areas where the surface has a uniform albedo. An experimental comparison of these two techniques should be performed to better quantify these effects. Combining structured light with the shape-from-rotation paradigm should result in an algorithm that works under a wider variety of conditions, but at the expense of increased hardware complexity.

An alternative to the approach presented in this paper is to use the silhouette of the object in each frame to construct a bounding volume for the object [Szeliski, 1990]. This bounding volume can provide a non-linear (inequality) constraint on the position of surface points. Tracking the silhouettes through three or more images can also be used to estimate the location and curvature of points on the limb of the object [Giblin and Weiss, 1987; Vaillant, 1990; Cipolla and Blake, 1990]. Combining silhouette-based and flow-based approaches should yield an algorithm that works for a much wider variety of object shapes and textures.

Our shape from rotation algorithm would be even more useful if we could change the position of the camera and/or the object. The former case is easier to handle: we simply re-calibrate the system, and continue processing with the new camera parameters. The change