Intrinsic Video (Supplementary Material)

Naejin Kong, Peter V. Gehler, and Michael J. Black

Max Planck Institute for Intelligent Systems, Tübingen, Germany
{naejin.kong,peter.gehler,black}@tuebingen.mpg.de

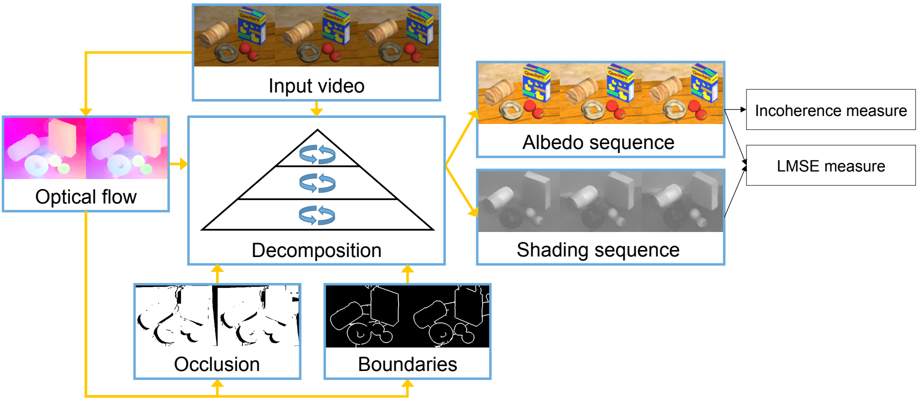


Fig. 1. Test procedure

In this supplementary material, we first show results on extra synthetic and real test sequences (Section 1). We also show that the method deals with violations of our assumptions; for example, non-Lambertian surfaces (Section 2). We perform a sensitivity analysis and find that the results are quite insensitive to the parameter settings (Section 3). We explain the error metrics used to compare our estimated intrinsic video to ground truth (Section 4). We next provide more details on our optimization scheme (Section 5). We then depict more examples of non-local weights that are important to improve shading estimation (Section 6). Finally, we explain how our input data was created, and show the full results in addition to optical flow, occlusion and boundary intrinsic images for each example (Section 7).

We overview our test procedure in Fig. 1: The input is a video sequence and optical flow estimated from the sequence. We used the Classic+NL method [5] with its default settings to compute the optical flow. Occlusion maps and motion boundary maps detected from the flow are used in our coarse-to-fine decomposition algorithm. The output is the estimated albedo and shading sequences. These sequences of optical flow, occlusion, motion boundaries, albedo and shading define *intrinsic video*. We measure an LMSE (local mean squared error) [3] of the reconstructed albedo and shading images, which is a standard error measure in the field. We also introduce a new measure of temporal incoherence, which assesses how consistent the albedo is over time. Details of the LMSE and incoherence metrics are given below in Sections 4.1 and 4.2, respectively.

Unless otherwise mentioned, all equations, figures and references indicate those in this document. Table 1 summarizes the abbreviations of different methods used in Figures 4-8 of the main paper.

Table 1. Abbreviations used in the main paper

Indicator	Meaning
IV w/o Flow	Our method without temporal coherence terms
IV	Our method using optical flow estimated by Classic+NL
IV w/ GT Flow	Our method using ground truth optical flow
CRET	Baseline color-Retinex algorithm in [3]
GS	More advanced Retinex-based method in [2]

1 Extra Synthetic and Real Examples

In Figures 2 and 3, we illustrate another synthetic example and another real example that were omitted from the main paper due to limited space. The results are consistent with those in the main paper. As shown in (g)-(r) of the figures, both of CRET and GS put lots of high-frequency albedo information into the shading image, and the albedo is overall inconsistent between frames. In contrast, our albedo image retains most details and the shading is piecewise smooth, mostly obeying object boundaries. Our recovered albedo is relatively more constant in time. As shown in (c)-(f) of the figures, our albedo produces less noisy flow fields, suggesting that our albedo has better temporal coherence than the others.

2 Violating our Assumptions

We modified our synthetic sequences by adding a specular component to some materials, and tested our method on the modified sequences. Our method still produced fairly good results as shown in Fig. 4. Shading is as accurate as before and albedo absorbs the specularities. Coherence of the reconstructed albedo sequence is clearly enhanced from the original images. Note that our real sequences already contain some specularities and shadows. The results on full frames are also shown in Figures 34, 35 and 36 .

3 Parameter Sensitivity

Our algorithm is relatively insensitive to significant variations in the parameters λ_D, λ_{T_A}, λ_{T_S}, λ_{S_s}, $\lambda_{S_{\text{cpl}}}$, λ_{S_m} and α. We perturbed each of these parameters

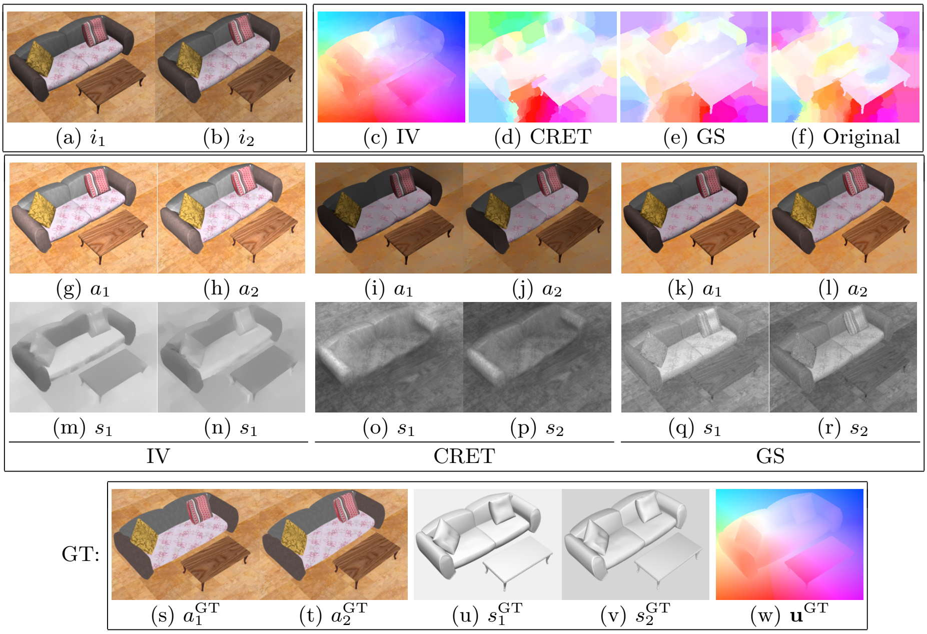
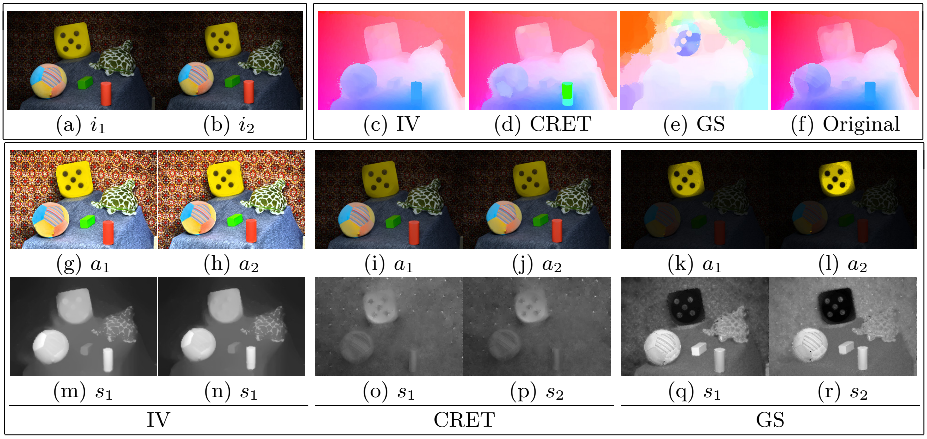


Fig. 2. Synthetic example in which the camera zooms in from the fourth frame, and illumination variation is more drastic than in the first synthetic example (Fig. 4 in the main paper). (a),(b) Two frames from the sequence. (c)-(e) Flow from the albedo estimated from our method (IV), CRET and GS. (f) Flow from the original images. (g)-(l) Albedo from IV, CRET and GS. (m)-(r) Shading from IV, CRET and GS. (s)-(w) Ground truth albedo, shading and flow.

up to 20% from the default setting, while fixing the other parameters as their default values. We used our three synthetic sequences and ground truth flow as input, and applied our method with the perturbed parameter values. For each perturbed parameter value, we measured an LMSE averaged over the three sequences and plotted a graph in Fig. 5. In the graph, the horizontal scale indicates the amount of perturbation from the default value (noted as 0%) of each parameter. The vertical scale has the same order of magnitude as that used in Fig. 6 (left) of the main paper. It shows no significant error variation. This suggests that our method is robust to changes in the parameter settings.

4 Error Metrics

In this section, we provide more details on error metrics used to compare our estimated intrinsic video to ground truth.



(a) i_1 (b) i_2 (c) IV (d) CRET (e) GS (f) Original

(g) a_1 (h) a_2 (i) a_1 (j) a_2 (k) a_1 (l) a_2

(m) s_1 (n) s_1 (o) s_1 (p) s_2 (q) s_1 (r) s_2

IV CRET GS

Fig. 3. Real example in which we introduced slowly varying illumination by mounting a continuous light source on top of the moving camera. (a),(b) Two frames from the sequence. (c)-(e) Flow from the albedo estimated from our method (IV), CRET and GS. (f) Flow from the original images. (g)-(l) Albedo from IV, CRET and GS. (m)-(r) Shading from IV, CRET and GS.

4.1 LMSE metric

The LMSE metric [3] measures an error for both estimated albedo and shading images compared to their ground truth at each frame. This metric is locally scale-invariant, and defined as

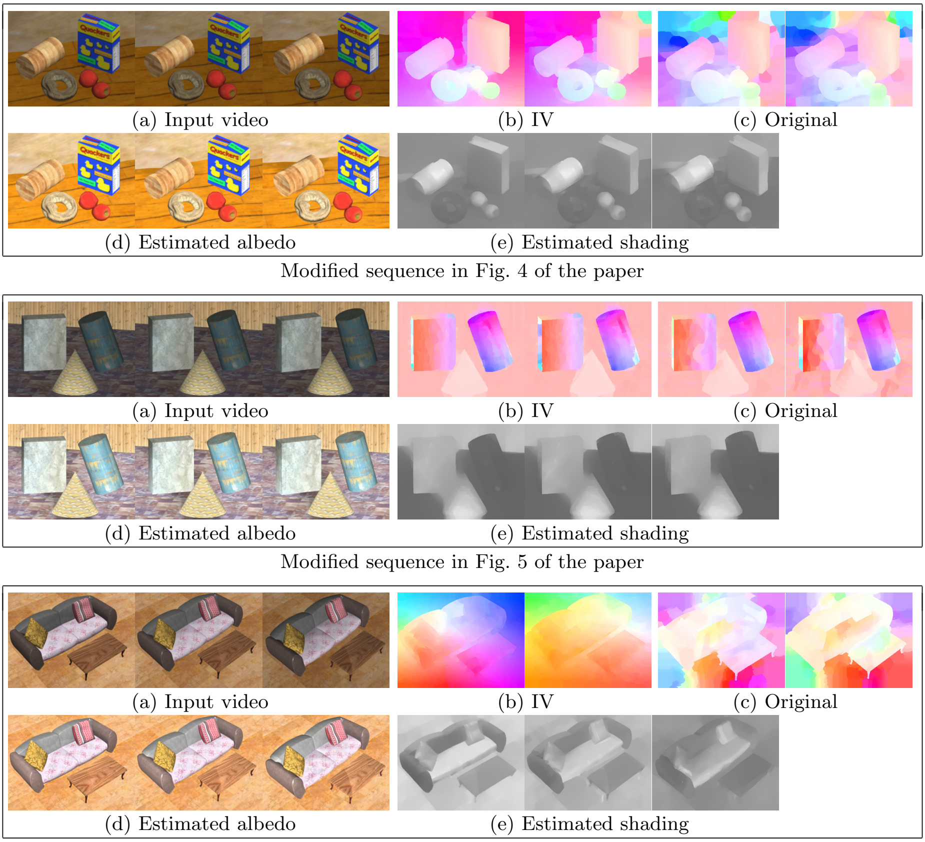
$$\mathrm{LMSE}(a^{\mathrm{GT}}, a^{\mathrm{EST}}, s^{\mathrm{GT}}, s^{\mathrm{EST}}) = \frac{1}{2}\frac{\mathrm{LMSE}_k(a^{\mathrm{GT}}, a^{\mathrm{EST}})}{\mathrm{LMSE}_k(a^{\mathrm{GT}}, 0)} + \frac{1}{2}\frac{\mathrm{LMSE}_k(s^{\mathrm{GT}}, s^{\mathrm{EST}})}{\mathrm{LMSE}_k(s^{\mathrm{GT}}, 0)}, \quad (1)$$

$$\mathrm{LMSE}_k(x^{\mathrm{GT}}, x^{\mathrm{EST}}) = \sum_{\mathbf{w} \in \mathrm{W}_k} \|x^{\mathrm{GT}}_{\mathbf{w}} - \hat{\alpha} x^{\mathrm{EST}}_{\mathbf{w}}\|^2 \quad (2)$$

where $\hat{\alpha} = \mathrm{argmin}_{\alpha} \|x^{\mathrm{GT}}_{\mathbf{w}} - \alpha x^{\mathrm{EST}}_{\mathbf{w}}\|^2$. Here, W_k is a set of $k \times k$ windows spaced in steps of $k/2$ (where $k = 20$), $x_{\mathbf{w}}$ is a vectorized window in the set W_k for image x. A superscript $^{\mathrm{GT}}$ or $^{\mathrm{EST}}$ indicates that the variable comes from the ground truth image or the estimated image, respectively. Note that the albedo image is composed of RGB channels; for each frame, we apply this error metric individually to each RGB channel and take the mean of those three errors. We average this LMSE over all frames and then average this over all three synthetic examples.

4.2 Incoherence metric

Optical flow methods typically assume brightness constancy, which is violated if illumination in the sequence is inconsistent over time. If one has accurate albedo estimation at every frame then *albedo constancy* should hold, making optical flow easy to estimate from the albedo. Since violations of constancy increase errors in



(a) Input video (b) IV (c) Original

(d) Estimated albedo (e) Estimated shading

Modified sequence in Fig. 4 of the paper

(a) Input video (b) IV (c) Original

(d) Estimated albedo (e) Estimated shading

Modified sequence in Fig. 5 of the paper

(a) Input video (b) IV (c) Original

(d) Estimated albedo (e) Estimated shading

Modified sequence in Fig. 2

Fig. 4. Adding violations of our assumptions. (a) Three frames from a sequence modified by adding a specular component to some materials in the original sequence. (b) Flow from the albedo estimated from our method. (c) Flow from the original images. (d) Estimated albedo. (e) Estimated shading.

optical flow, the optical flow error provides a measure of how constant a sequence is in time. Note that while we assume albedo in the world is constant, in a video sequence the albedo values are moving and this movement is described by the optical flow. Evaluating the optical flow accuracy has the nice property of being directly relevant to a task. If intrinsic video estimation can improve optical flow accuracy, then this could be immediately beneficial to optical flow algorithms.

The implementation of the Classic+NL method has a setting to use the standard brightness constancy assumption rather than a more complex model assuming constancy of a texture decomposition. We apply Classic+NL with brightness instead of texture decomposition to each of the sequences reconstructed from our method, previous methods, and the original images. We then compute an error of each flow field compared to the ground truth flow using EPE (averaged end-

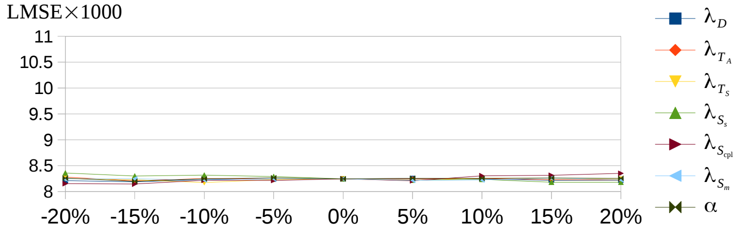


Fig. 5. Parameter sensitivity. We perturbed each of λ_D, λ_{T_A}, λ_{T_S}, λ_{S_s}, $\lambda_{S_{\mathrm{cpl}}}$, λ_{S_m} and α from their default values and measured LMSEs averaged over the three synthetic sequences. The horizontal scale indicates the amount of perturbation from the default parameter value. The graph shows no significant error variation.

point-error) [1] defined as follows:

$$\mathrm{EPE}(\mathbf{u}^{\mathrm{GT}}, \mathbf{u}^{\mathrm{EST}}) = \frac{1}{N} \sum_{\mathbf{x}} \sqrt{\|\mathbf{u}^{\mathrm{GT}}(\mathbf{x}) - \mathbf{u}^{\mathrm{EST}}(\mathbf{x})\|^2}, \tag{3}$$

where N is the number of pixels, $\mathbf{x}$ is pixel location, $\mathbf{u}(\mathbf{x})$ is an optical flow vector at $\mathbf{x}$, $\mathbf{u}^{\mathrm{GT}}$ is ground truth optical flow, and $\mathbf{u}^{\mathrm{EST}}$ is estimated optical flow. This provides a measure of how temporally coherent the albedo sequence is; a more coherent sequence produce lower EPE. Note that our input optical flow, computed with the Classic+NL method using its default settings, is not a part of this incoherence measure since it uses a different texture-decomposition constancy assumption. We average this EPE over all frames and then average it over the three synthetic examples.

We illustrate the flow from this incoherence measure in (c)-(f) of Figures 4, 5, 7 and 8 of the main paper as well as Figures 2 and 3. Each flow field is visualized by using the standard color coding for its direction and magnitude. This flow image provides an intuitive visualization of coherence, since an albedo sequence with better temporal coherence will produce flow images that look closer to the images of ground truth flow.

5 Optimization Details

We give more details on our optimization scheme here. To minimize our objective function, Eq. (2) of the main paper, we adopt a coarse to fine pyramid-based approach and incremental update scheme similar in spirit to the flow estimation method in [5].

Coarse to find approach. We use a 3-level Gaussian pyramid with a scale factor of 2 for all our video input (the resolution is either 320 × 240 or 320 × 214). At the coarsest pyramid level, we start by setting all unknowns to 0. At each pyramid level, we incrementally update the unknowns, rather than directly estimate the variables, as explained below. At each finer level in the pyramid, the shading values estimated at a coarser level are up-sampled, then the log difference from input video frame is taken to initialize the albedo values at that level.

Incremental update. At each pyramid level, the optimization problem in Eq. (2) of the main paper is solved repeatedly but incrementally. This scheme was proven to be a best practice for flow estimation in [5]. The variables at each iteration k are defined as

$$A_t^k = A_t^{k-1} + \hat{A}_t^k \text{ and } S_t^k = S_t^{k-1} + \hat{S}_t^k,$$

where A_t^{k-1} and S_t^{k-1} are the albedo and shading estimated at the last iteration, $k-1$, $\hat{A}_t^k$ and $\hat{S}_t^k$ are incremental variables to be estimated at current iteration k. Then, the objective function to minimize is now

$$\begin{aligned} \underset{\{\hat{A}_t^k, \hat{S}_t^k, \tilde{S}_t^k\}}{\operatorname{argmin}} \ & \sum_t f_D(A_t^k, S_t^k | I_t) + f_{T_A}(A_{t+1}^k, A_t^k | \mathbf{u}_t) + f_{T_S}(A_{t+1}^k, A_t^k | \mathbf{u}_t) \\ & + f_A(A_t^k) + f'_S(S_t^k, \tilde{S}_t^k), \end{aligned} \tag{4}$$

where f'_S is the modified spatial shading term with the coupling variable $\tilde{S}_t^k$ in Eq. (13) of the main paper. In practice, we alternate between the estimation of $\{A_t^k, S_t^k\}$ and $\tilde{S}_t^k$ while encouraging the solutions to be similar with the quadratic coupling term. We iterate the incremental update 10 times.

6 Non-local Weights

The non-local term in our spatial shading prior (Eq. (9) in the main paper) assists spatial shading estimation, by preventing smoothing across motion boundaries. The non-local weights (Eq. (10) in the main paper) of the prior prevent spatial smoothing across motion boundaries, which are extracted from the optical flow. We show more examples on the non-local weights in Fig. 6.

7 Datasets and Full Results

Since our problem deals with moving camera/objects and varying illumination in the scene, none of existing datasets exactly fit in our requirements. We therefore generated three synthetic video sequences and three real video sequences, by varying different aspects of the motion and illumination.

7.1 Synthetic examples

Each of our synthetic video sequences includes ground truth values of albedo, shading, optical flow and occlusion. In order to obtain physically correct pixel

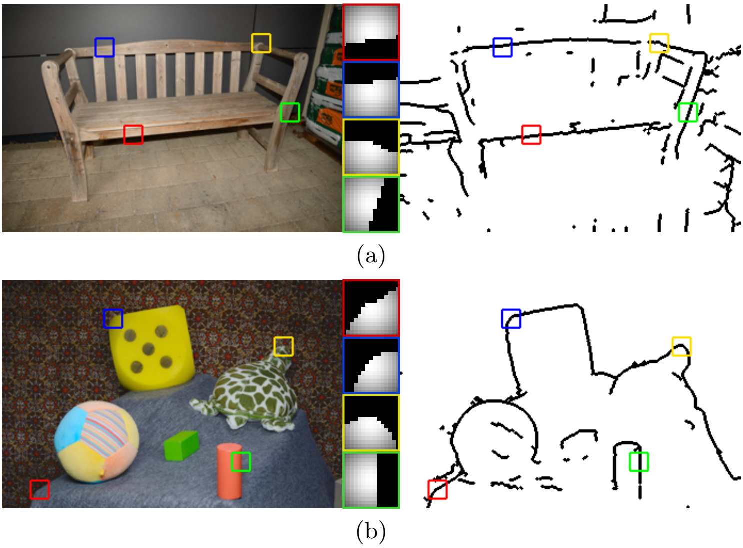


Fig. 6. Examples on non-local weights (Eq. (10) in the main paper). (a),(b) The weights computed from optical flow of the real sequence in Fig. 7 of the main paper and another real sequence in Fig. 3, respectively. In each of (a) and (b), the left image shows the first video frame, small boxes in the middle visualize 15 × 15 weights corresponding to the regions marked on the left image, and the right image shows motion boundaries detected from the optical flow, visualized as $1 - w_{\mathbf{u}_t}^{\mathrm{bnd}}$. (The left image in (b) is gamma-corrected for better visual presentation.)

values for the albedo and shading images, we used 3D rendering software Maya® with the Mental Ray® renderer. A Lambertian shader was assigned to all surfaces. We placed a white ambient light source and a white directional light source in the scene, where the directional source slowly but randomly changes its direction throughout the frames.

The ground truth optical flow and occlusion were computed as follows: At each frame, we cast a ray from each pixel to the scene surfaces and find the intersection point. In the next frame, we get the pixel location projected back from to the intersection point. Finally, we calculate the different between the two pixel locations. If the back-projected ray from the intersection point is blocked by other surfaces or goes beyond the image plane in the next frame, we mark the pixel in our occlusion maps. This is similar to the implementation in [4] but we extended it to deal with moving objects as well.

For our first synthetic example, its input data and ground truth values are shown in Fig. 7. Occlusion and motion boundary maps detected from optical flow are shown in Fig. 8. Note that these are three types of intrinsic video. Two other types of intrinsic video are the sequences of albedo and shading, and their reconstruction from our method is compared with those from previous methods in Fig. 9. Fig. 10 shows that our temporal coherence terms clearly improve albedo and shading estimation. Fig. 11 visually compares coherence of our reconstructed albedo with that from previous methods. Similarly, the second synthetic example is shown in Figures 12, 13, 14, 15, and 16, and the third synthetic example is shown in Figures 17, 18, 19, 20, and 21. Note that the CRET [3] and GS [2] methods are applied to each frame of the video independently. When our method is applied without temporal coherence terms, the spatial shading prior in Eq. (9) of the main paper uses only the local term.

Each example involves a different type of motion and illumination variation (see captions in Figures 7, 12, 17 for details). Overall, both of CRET and GS put too much high-frequency albedo information into the shading image, and the albedo changes significantly from frame to frame. In contrast, our albedo sequence retains most details and the shading sequence is piecewise smooth, mostly obeying object boundaries. Our albedo sequence is more consistent in time than that from previous methods, and it could be even more consistent than the original video. This suggests that intrinsic video may be useful to improve optical flow estimation.

7.2 Real examples

We captured real video sequences by serially taking photographs using a commercial DSLR camera (Nikon® D600) with a flash light or continuous lights. Our real examples involve different types of motion and illumination variation, corresponding to those in the synthetic examples.

Input data for our first real example is shown in Fig. 22, along with the optical flow, occlusion and motion boundary intrinsic images. The reconstructed albedo and shading sequences from ours and previous methods are compared in Fig. 23. Fig. 24 shows that the our temporal coherence terms clearly improve albedo and shading estimation. Fig. 25 visually compares coherence of our reconstructed albedo with that from previous methods. Similarly, the second real example is shown in Figures 26, 27, 28, and 29, and the third real example is shown in Figures 30, 31, 32, and 33.

Each example involves a different type of motion and illumination variation (see captions in Figures 22, 26, 30 for details). The results are consistent with those on synthetic sequences. Visually, our method significantly outperforms the previous methods. The shading from previous methods carries a lot of albedo information, but our shading sequence has few albedo details and well captures the overall shape of the scene. Previous methods sometime almost completely miss the shape of the scene in their shading images, and the albedo is overall inconsistent between frames. Our albedo is very consistent in time and keeps most details. Our shading well captures overall shape of the scene and presents very few albedo details. Our method produces a lot cleaner but less noisy flow fields than previous methods; coherence of our albedo could be even better than that of the original video.

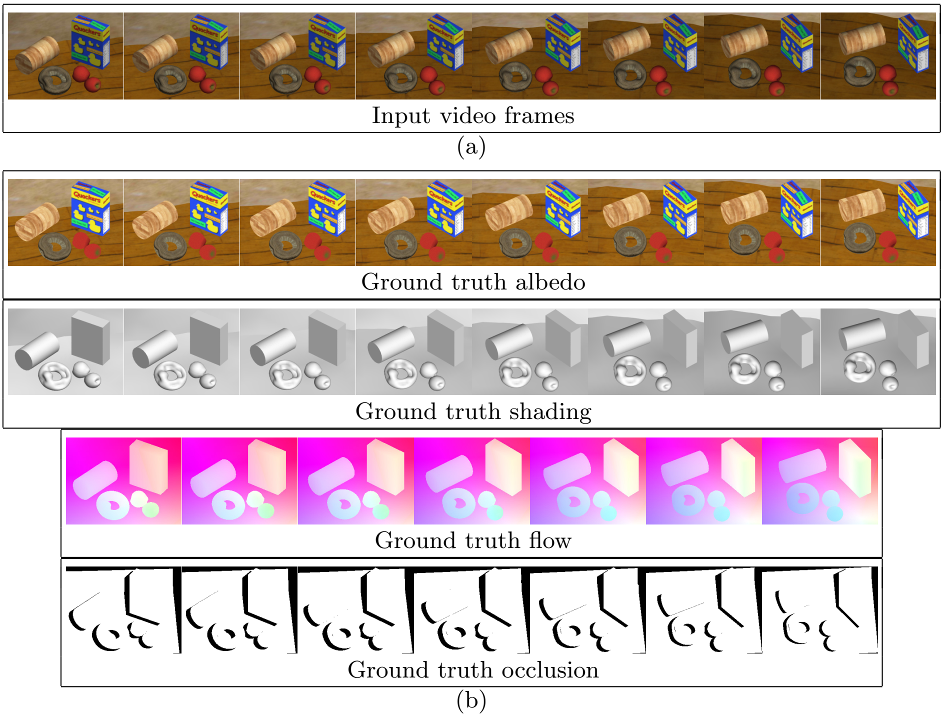
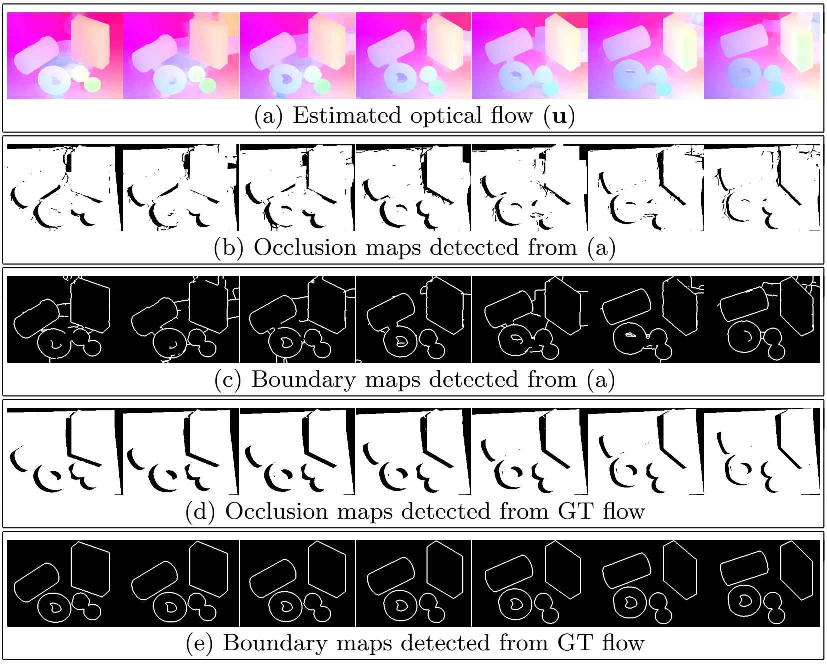


Fig. 7. (a) Input video (8 frames; 320 × 240): in this example, a camera is freely moving and illumination varies significantly over time. (b) Ground truth albedo, shading, optical flow, and occlusion for the input video.



(a) Estimated optical flow ($\mathbf{u}$)

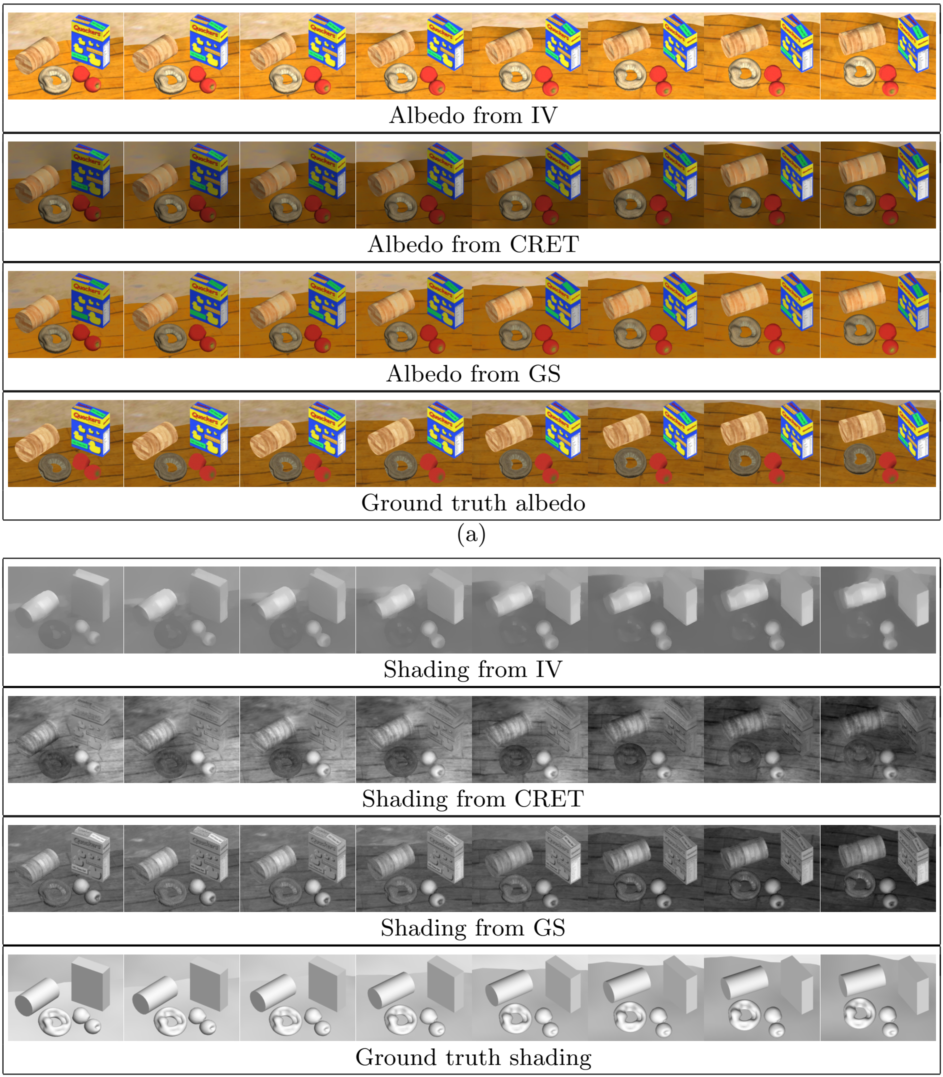
(b) Occlusion maps detected from (a)

(c) Boundary maps detected from (a)

(d) Occlusion maps detected from GT flow

(e) Boundary maps detected from GT flow

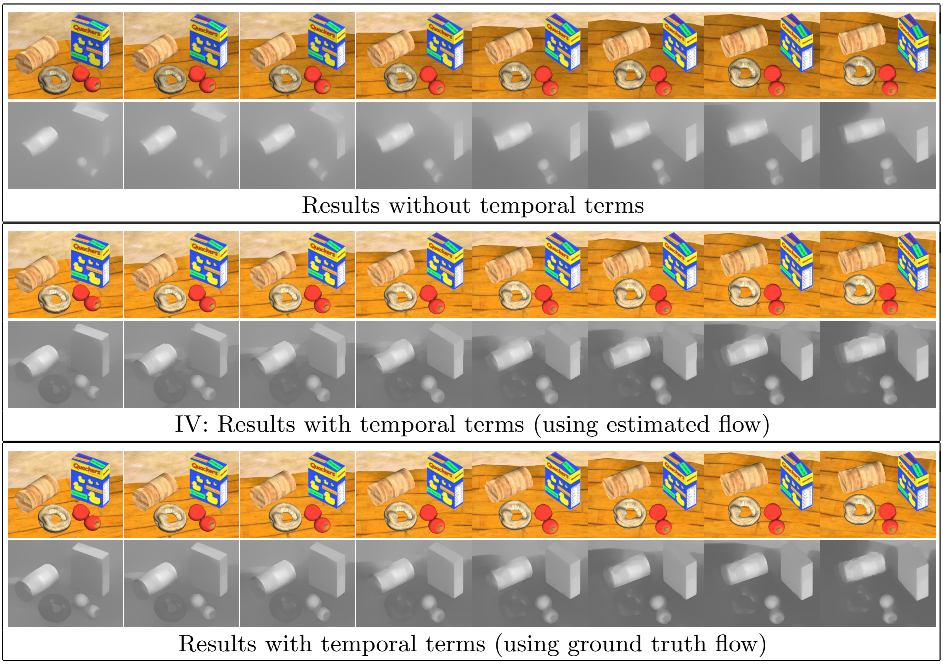
Fig. 8. (a) Optical flow computed from the video in Fig. 7(a) using default Classic+NL [5]. (b) Occlusion maps ($w_{\mathbf{u}}^{\text{occ}}$) detected from (a). (c) Motion boundary maps ($w_{\mathbf{u}}^{\text{bnd}}$) detected from (a). (d) Occlusion maps ($w_{\mathbf{u}_{\text{GT}}}^{\text{occ}}$) detected from ground truth flow. (e) Motion boundary maps ($w_{\mathbf{u}_{\text{GT}}}^{\text{bnd}}$) detected from ground truth flow.



(a)

(b)

Fig. 9. (a) Albedo estimated by our method (IV), CRET [3], GS [2], and the ground truth albedo. (b) Shading estimated by IV, CRET, GS, and the ground truth shading. Both of CRET and GS put too much high-frequency albedo information into the shading image. Also the albedo changes significantly from frame to frame. In contrast, our albedo image retains most details and the shading is piecewise smooth, mostly obeying object boundaries.



Results without temporal terms

IV: Results with temporal terms (using estimated flow)

Results with temporal terms (using ground truth flow)

Fig. 10. Albedo and shading with and without our temporal terms. We also compare the results of using the estimated flow versus using ground truth flow. The comparison shows that our temporal terms clearly improve albedo and shading estimation, especially the shading estimation. Second, temporal coherence provided by the computed optical flow is good enough to produce results similar to those estimated with the ground truth optical flow.

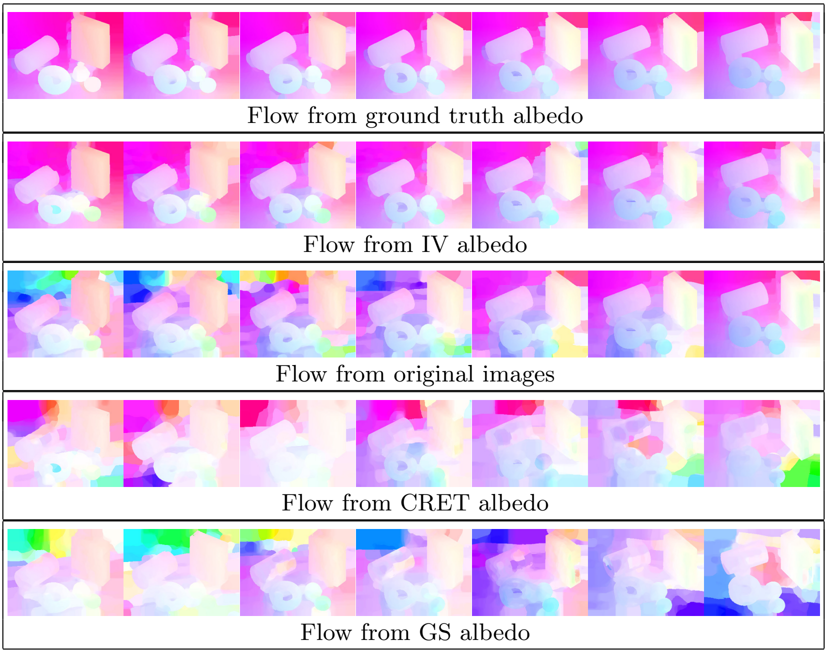


Fig. 11. Flow from ground truth albedo, IV (our method) albedo, original images, CRET albedo, and GS albedo. Our albedo is clearly more consistent than the albedo sequence estimated by previous methods. In addition, note that our albedo sequence is more consistent than the original video.

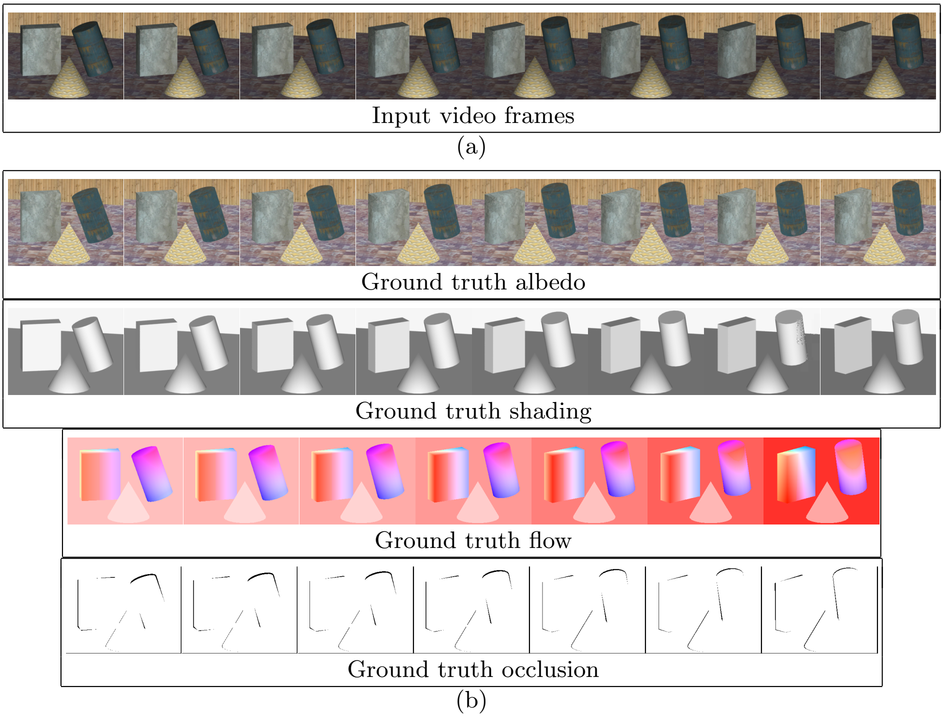
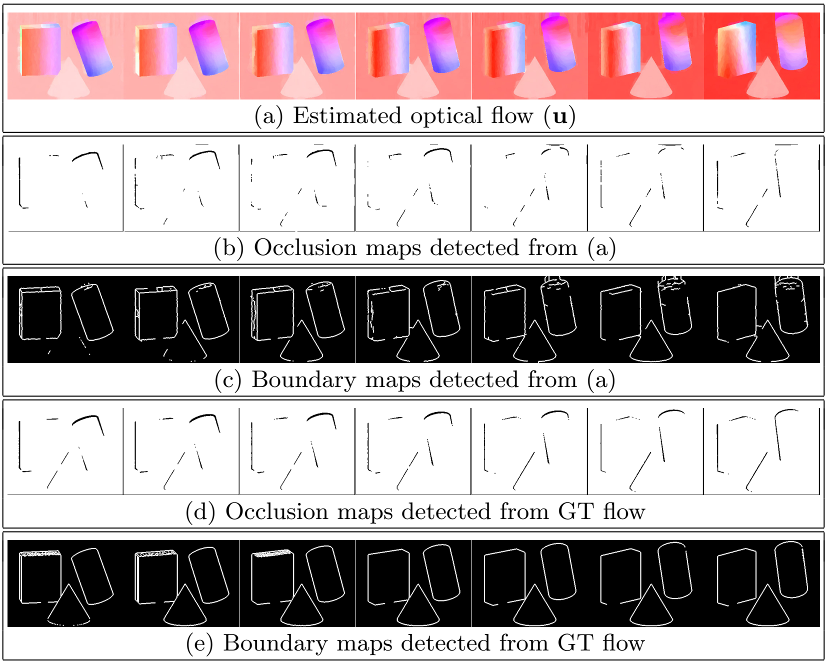


Fig. 12. (a) Input video (8 frames; 320×240): in this example, all objects in the scene are moving while the camera translates. Illumination does not change much in this case. (b) Ground truth albedo, shading, optical flow, and occlusion for the input video.



(a) Estimated optical flow ($\mathbf{u}$)

(b) Occlusion maps detected from (a)

(c) Boundary maps detected from (a)

(d) Occlusion maps detected from GT flow

(e) Boundary maps detected from GT flow

Fig. 13. (a) Optical flow computed from the video in Fig. 12(a) using default Classic+NL [5]. (b) Occlusion maps ($w_{\mathbf{u}}^{\text{occ}}$) detected from (a). (c) Motion boundary maps ($w_{\mathbf{u}}^{\text{bnd}}$) detected from (a). (d) Occlusion maps ($w_{\mathbf{u}_{\text{GT}}}^{\text{occ}}$) detected from ground truth flow. (e) Motion boundary maps ($w_{\mathbf{u}_{\text{GT}}}^{\text{bnd}}$) detected from ground truth flow.

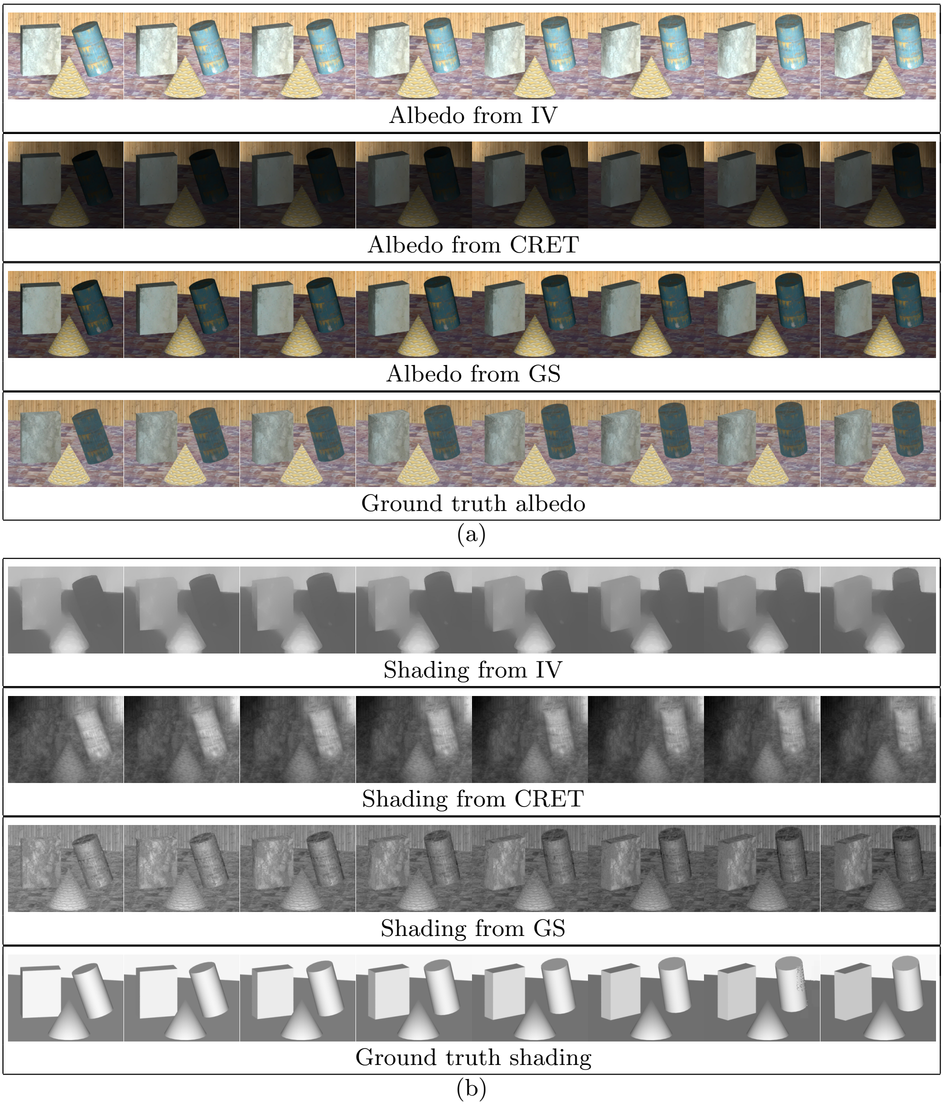
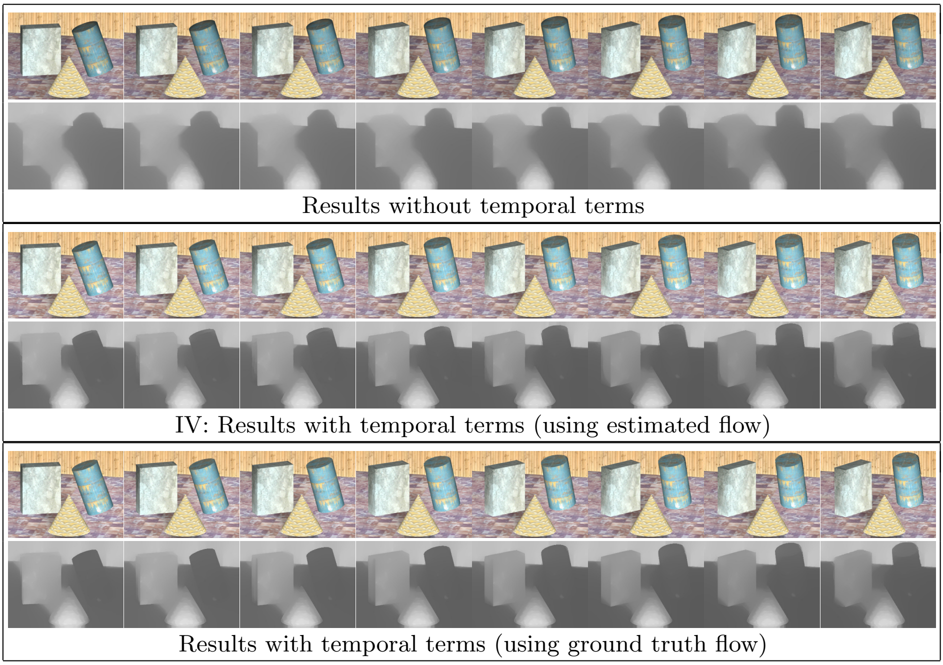


Fig. 14. (a) Albedo estimated by our method (IV), CRET [3], GS [2], and the ground truth albedo. (b) Shading estimated by IV, CRET, GS, and the ground truth shading. The albedo estimated by CRET or NIPS misses lots of details, which are carried in the shading incorrectly. In contrast, our albedo keeps most details and its shading presents very few albedo details in it.

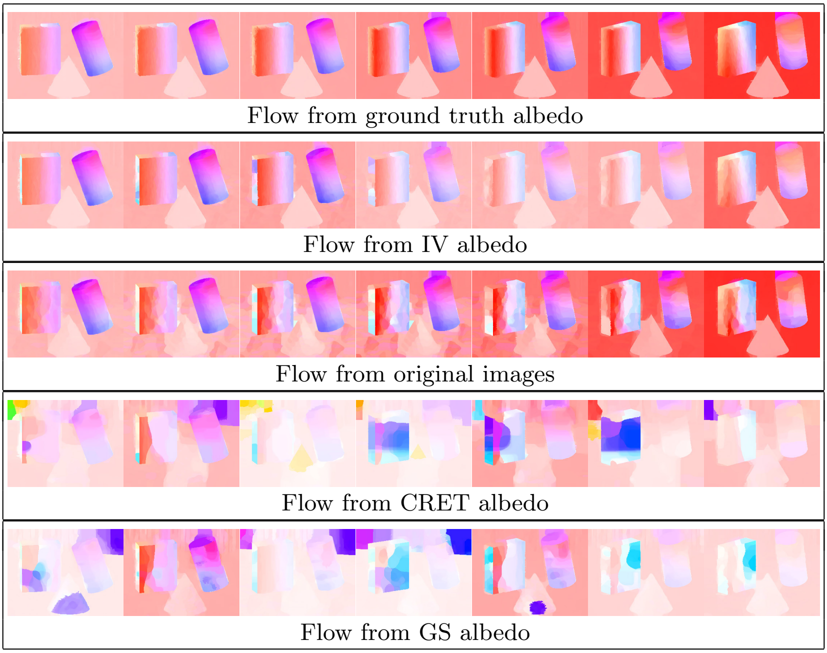


Results without temporal terms

IV: Results with temporal terms (using estimated flow)

Results with temporal terms (using ground truth flow)

Fig. 15. Albedo and shading with and without our temporal terms. We also compare the results of using the estimated flow versus using ground truth flow. This comparison shows that our temporal coherence terms improve albedo and shading estimation. In addition, temporal coherence provided by the computed optical flow is good enough to produce results similar to those estimated with the ground truth optical flow.



Flow from ground truth albedo

Flow from IV albedo

Flow from original images

Flow from CRET albedo

Flow from GS albedo

Fig. 16. Flow from ground truth albedo, IV (our method) albedo, original images, CRET albedo, and GS albedo. Our albedo is more consistent than that from previous methods. The flow fields from our albedo are cleaner than those from the original video, and even close to those from ground truth albedo.

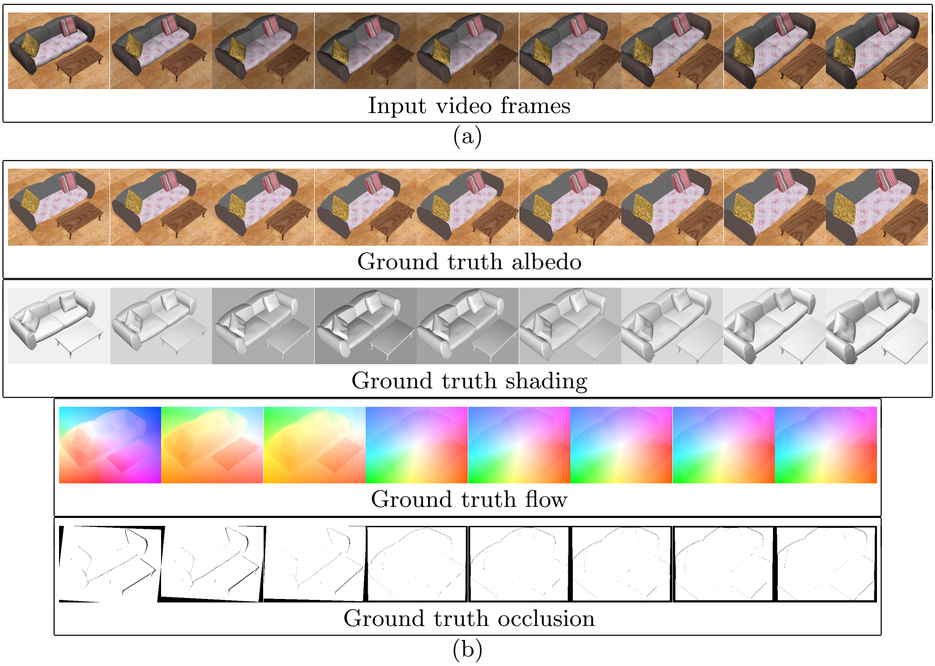
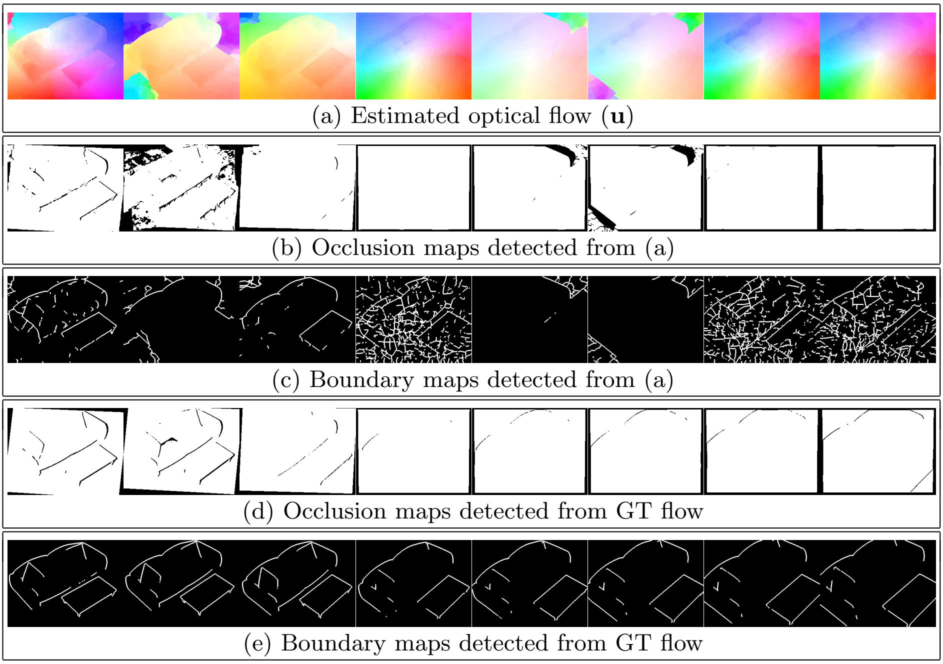


Fig. 17. (a) Input video (9 frames; 320 × 240): in this example, the camera zooms in from the fourth frame, and illumination variation is more drastic than in the previous example. (b) Ground truth albedo, shading, optical flow, and occlusion for the input video.


(a) Estimated optical flow ($\mathbf{u}$)

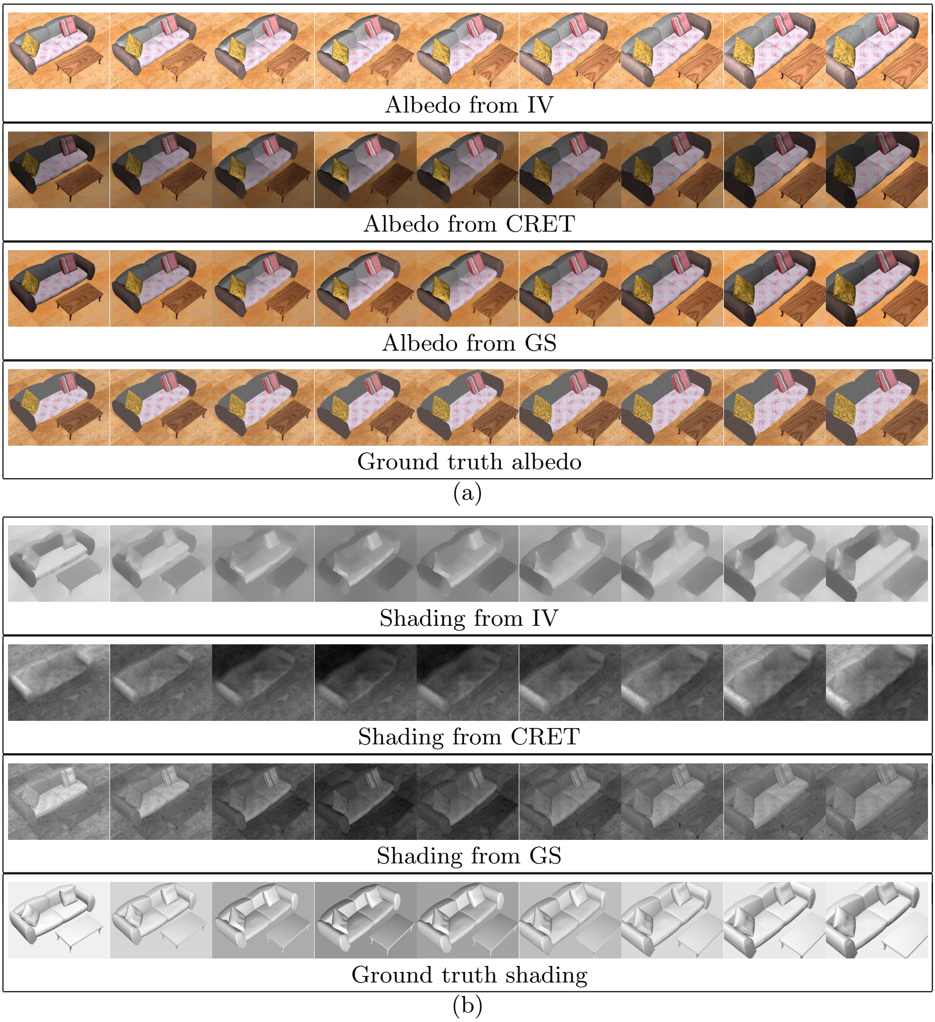
(b) Occlusion maps detected from (a)

(c) Boundary maps detected from (a)

(d) Occlusion maps detected from GT flow

(e) Boundary maps detected from GT flow

Fig. 18. (a) Optical flow computed from the video in Fig. 17(a) using default Classic+NL [5]. (b) Occlusion maps ($w_{\mathbf{u}}^{\text{occ}}$) detected from (a). (c) Motion boundary maps ($w_{\mathbf{u}}^{\text{bnd}}$) detected from (a). (d) Occlusion maps ($w_{\mathbf{u}_{\text{GT}}}^{\text{occ}}$) detected from ground truth flow. (e) Motion boundary maps ($w_{\mathbf{u}_{\text{GT}}}^{\text{bnd}}$) detected from ground truth flow. In this case, estimated optical flow is rather noisy and the motion boundary maps have been affected by that. However, accurate flow consistently yields clean maps as in (d) and (e). Our method using (b) and (c) still produces high-quality albedo and shading as shown in Fig. 19.



Albedo from IV

Albedo from CRET

Albedo from GS

Ground truth albedo

(a)

Shading from IV

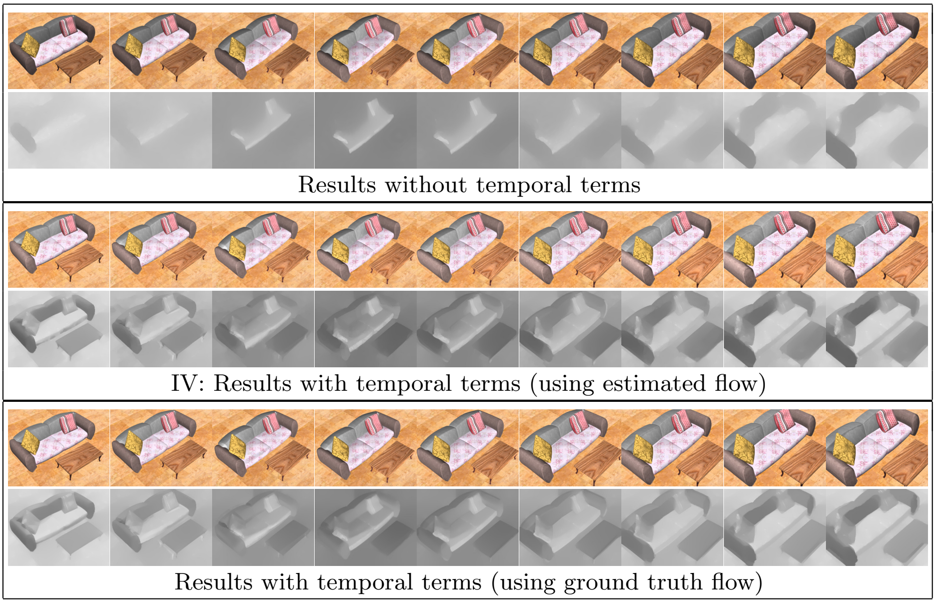
Shading from CRET

Shading from GS

Ground truth shading

(b)

Fig. 19. (a) Albedo estimated by our method (IV), CRET [3], GS [2], and the ground truth albedo. (b) Shading estimated by IV, CRET, GS, and the ground truth shading. The albedo estimated by previous methods misses many details and their shading images contain lots of albedo information. Our shading presents very few albedo details and is piecewise smooth while mostly obeying object boundaries.



Results without temporal terms

IV: Results with temporal terms (using estimated flow)

Results with temporal terms (using ground truth flow)

Fig. 20. Albedo and shading with and without our temporal terms. We also compare the results of using the estimated flow versus using ground truth flow. These results show that our temporal coherence terms improve albedo and shading estimation. In particular, temporal coherence improves the precision of the shading boundaries between surfaces. In addition, temporal coherence provided by the computed optical flow is good enough to produce results similar to those estimated with the ground truth optical flow.

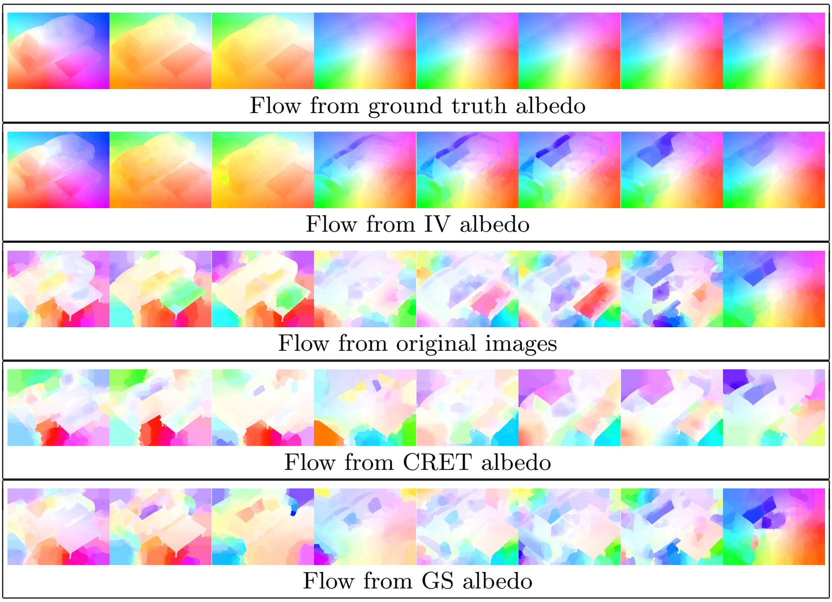
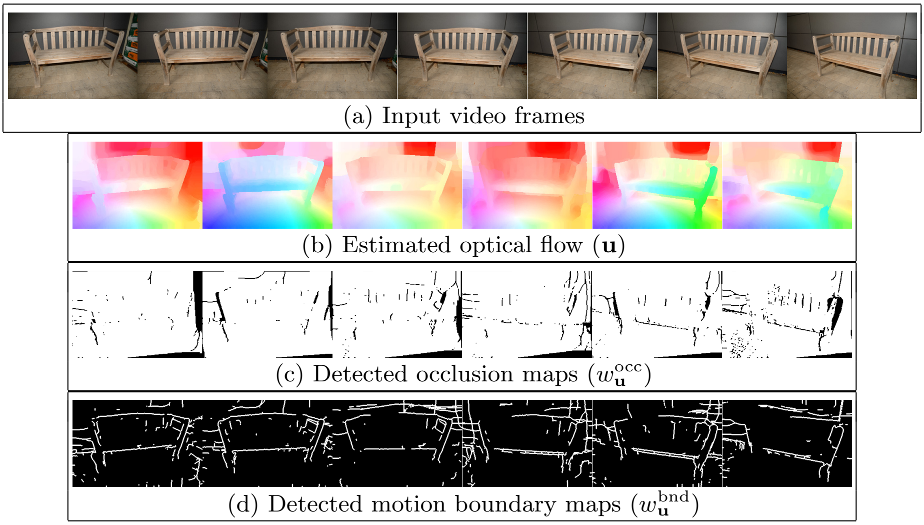


Fig. 21. Flow from ground truth albedo, IV (our method) albedo, original images, CRET albedo, and GS albedo. Our albedo sequence is clearly more consistent than the albedo sequence estimated by previous methods. In addition, note that our albedo sequence is more consistent than the original video.



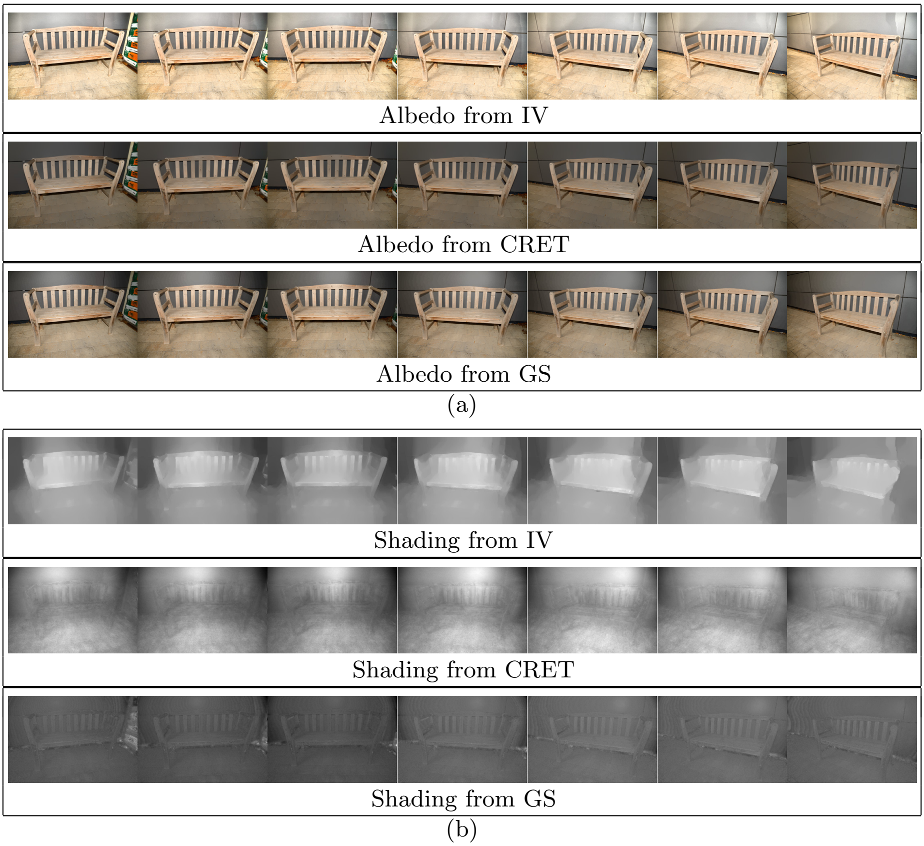
(a) Input video frames

(b) Estimated optical flow ($\mathbf{u}$)

(c) Detected occlusion maps ($w_{\mathbf{u}}^{\mathrm{occ}}$)

(d) Detected motion boundary maps ($w_{\mathbf{u}}^{\mathrm{bnd}}$)

Fig. 22. (a) Input video (7 frames; 320 × 214): in this example, the input video captures a static outdoor scene with a freely moving camera. A flashlight on top of the camera was used to vary illumination over time fairly drastically. (b) Optical flow computed from the video using default Classic+NL [5]. (c) Occlusion maps ($w_{\mathbf{u}}^{\mathrm{occ}}$) detected from (b). (d) Motion boundary maps ($w_{\mathbf{u}}^{\mathrm{bnd}}$) detected from (b).



Albedo from IV

Albedo from CRET

Albedo from GS

(a)

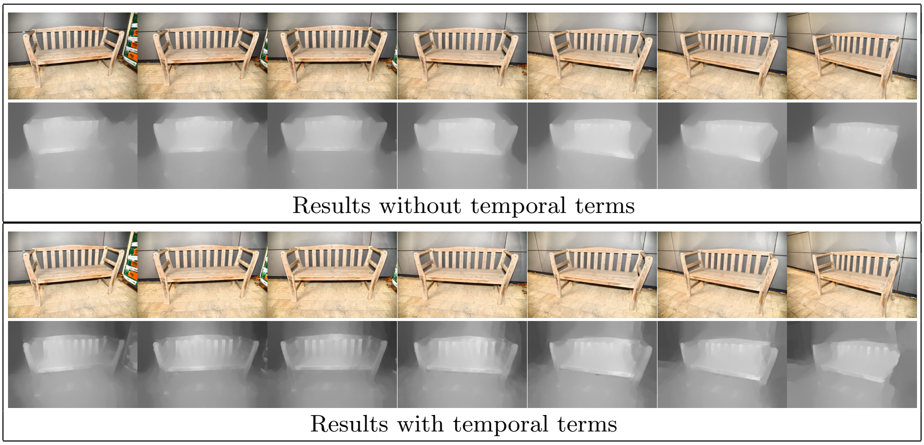
Shading from IV

Shading from CRET

Shading from GS

(b)

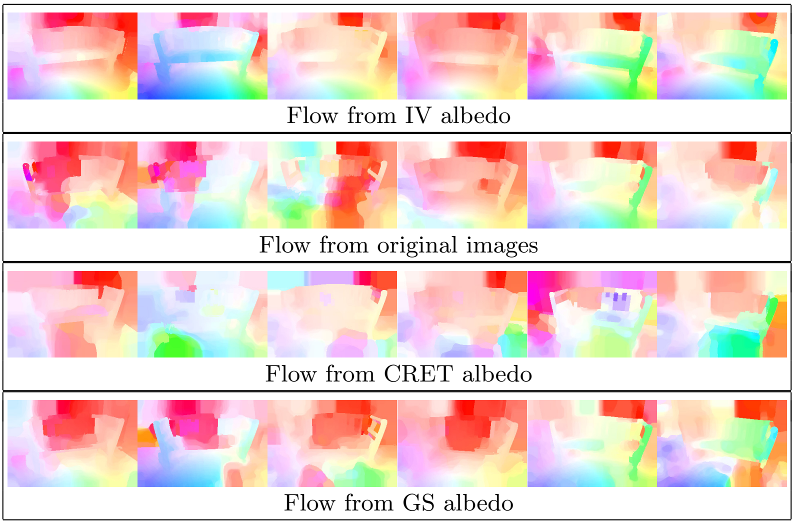
Fig. 23. (a) Albedo estimated by our method (IV), CRET [3] and GS [2]. (b) Shading estimated by IV, CRET and GS. Our method significantly outperforms previous methods. The shading from previous methods carries a lot of albedo information. In contrast, our shading sequence has few albedo details and well captures the overall shape of the scene, mostly obeying object boundaries.



Results without temporal terms

Results with temporal terms

Fig. 24. Albedo and shading estimated without temporal terms, and with temporal terms (using estimated flow). Here we see that using the temporal constraints on albedo and shading is important to getting sharper shading boundaries. These results show that our temporal coherence terms improve albedo and shading estimation.



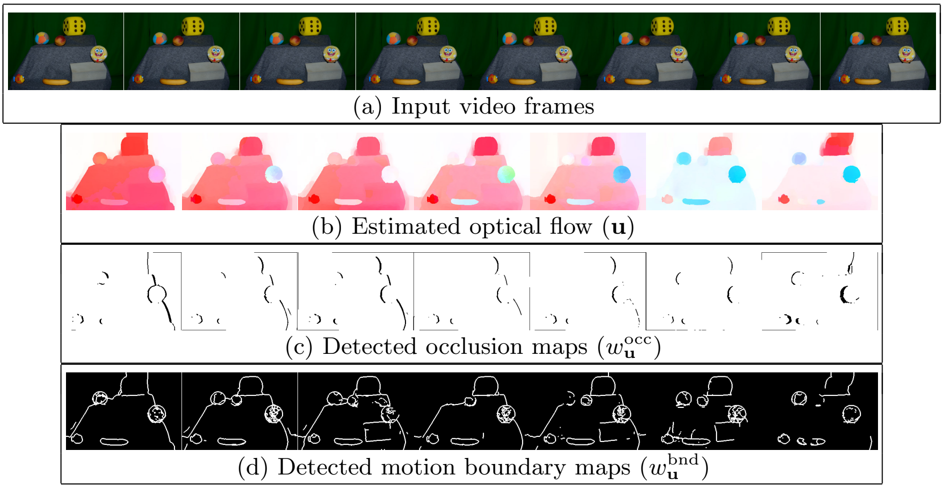
Flow from IV albedo

Flow from original images

Flow from CRET albedo

Flow from GS albedo

Fig. 25. Flow from IV (our method) albedo, original images, CRET albedo, and GS albedo as coherence visualization. While there is no ground true flow for this sequence, our reconstructed albedo produces less noisy flow fields, suggesting that our albedo has better temporal coherence than the others.

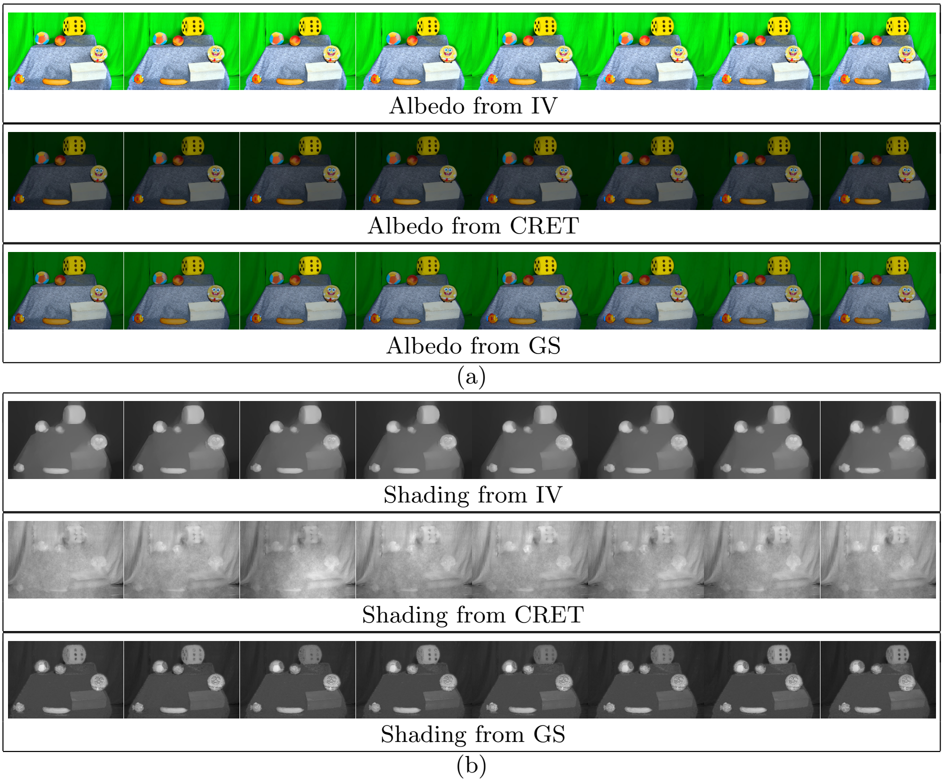

(a) Input video frames

(b) Estimated optical flow ($\mathbf{u}$)

(c) Detected occlusion maps ($w_{\mathbf{u}}^{\text{occ}}$)

(d) Detected motion boundary maps ($w_{\mathbf{u}}^{\text{bnd}}$)

Fig. 26. Input video (8 frames; 320 × 214): in this example, all objects continuously move but the background stays still. The camera and light sources are fixed. (b) Optical flow computed from the video using default Classic+NL [5]. (c) Occlusion maps ($w_{\mathbf{u}}^{\text{occ}}$) detected from (b). (d) Motion boundary maps ($w_{\mathbf{u}}^{\text{bnd}}$) detected from (b).



Albedo from IV

Albedo from CRET

Albedo from GS

(a)

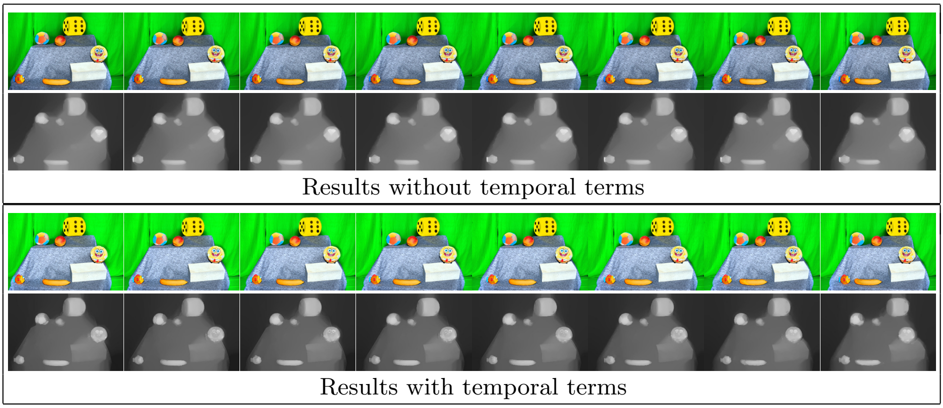
Shading from IV

Shading from CRET

Shading from GS

(b)

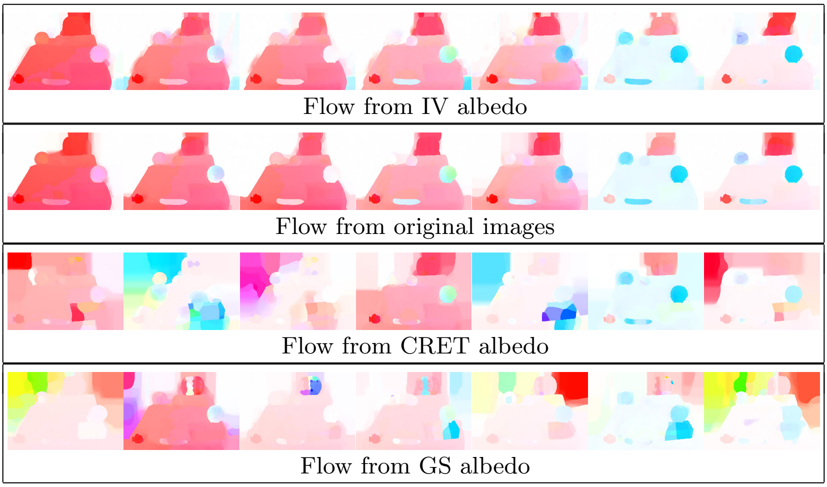
Fig. 27. (a) Albedo estimated by our method (IV), CRET [3] and GS [2]. (b) Shading estimated by IV, CRET and GS. The shading from CRET almost completely misses the shape of the scene, and the albedo from GS is inconsistent between frames (see the color of the cube). Our albedo is very consistent in time and our shading well captures overall shape of the scene.



Results without temporal terms

Results with temporal terms

Fig. 28. Albedo and shading estimated without temporal terms, and with temporal terms (using estimated flow). Here we see that using the temporal constraints on albedo and shading is important to getting sharper shading boundaries. These results show that our temporal coherence terms improve albedo and shading estimation.



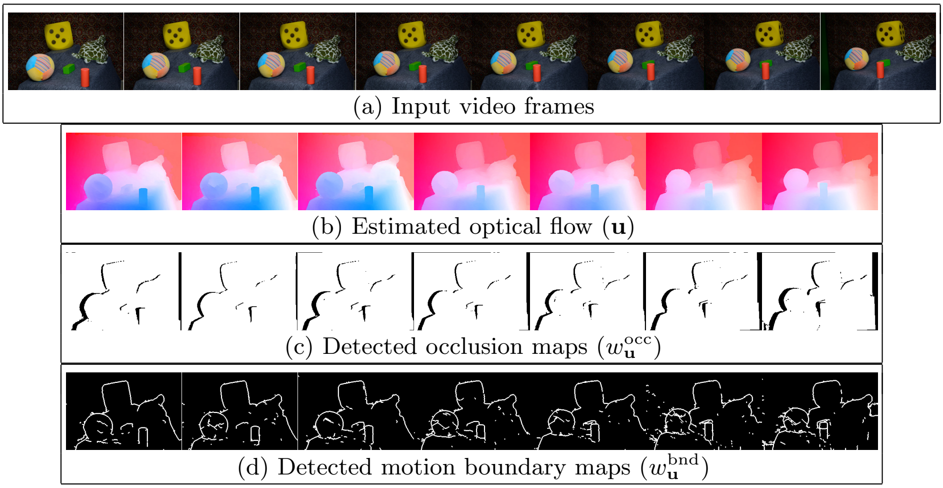
Flow from IV albedo

Flow from original images

Flow from CRET albedo

Flow from GS albedo

Fig. 29. Flow from IV (our method) albedo, original images, CRET albedo, and GS albedo as coherence visualization. Our albedo sequence is more coherent than the albedo sequence estimated by previous methods. In this case, since the illumination did not change, the original video is as coherent as our albedo sequence.



(a) Input video frames

(b) Estimated optical flow ($\mathbf{u}$)

(c) Detected occlusion maps ($w_{\mathbf{u}}^{\mathrm{occ}}$)

(d) Detected motion boundary maps ($w_{\mathbf{u}}^{\mathrm{bnd}}$)

Fig. 30. Input video (8 frames; 320×214): in this example, we introduced slowly varying illumination by mounting a continuous light source on top of the moving camera. (b) Optical flow computed from the video using default Classic+NL [5]. (c) Occlusion maps ($w_{\mathbf{u}}^{\mathrm{occ}}$) detected from (b). (d) Motion boundary maps ($w_{\mathbf{u}}^{\mathrm{bnd}}$) detected from (b).

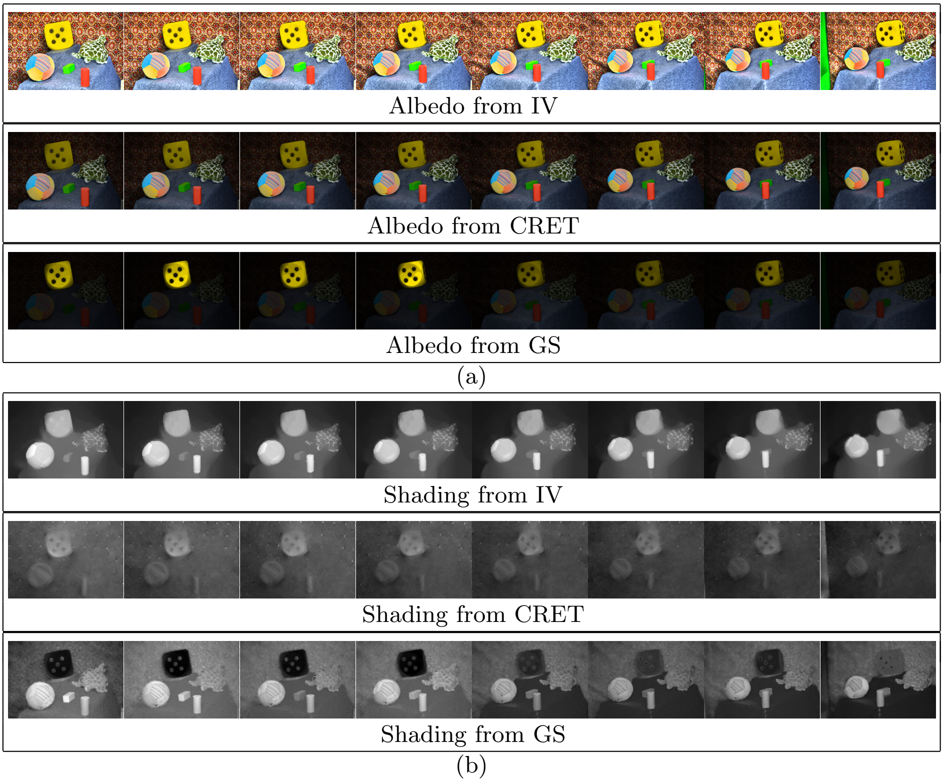
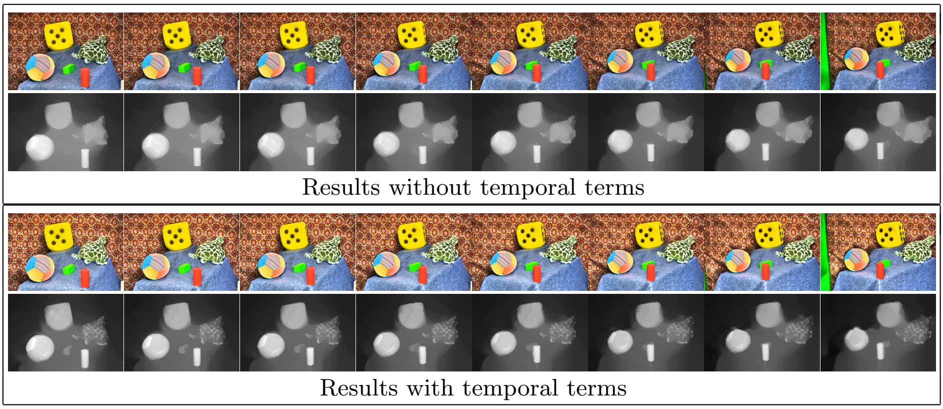


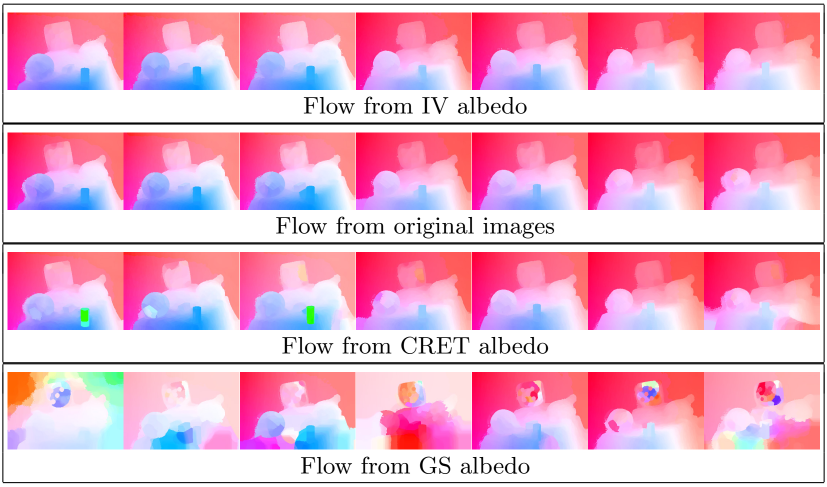
Fig. 31. (a) Albedo estimated by our method (IV), CRET [3] and GS [2]. (b) Shading estimated by IV, CRET and GS. The results are consistent with those on synthetic sequences. Our method again significantly outperforms previous ones. The shading from CRET almost completely misses the shape of the scene, and the albedo from GS is inconsistent between frames. On the other hand, our method produced favorable decomposition of albedo and shading even in this challenging example.



Results without temporal terms

Results with temporal terms

Fig. 32. Albedo and shading estimated without temporal terms, and with temporal terms (using estimated flow). Results are good even without the temporal terms, but the green block is almost missing in the shading images. With the temporal terms, the shading is improved to capture the shape of the block. These results show that our temporal coherence terms improve albedo and shading estimation.



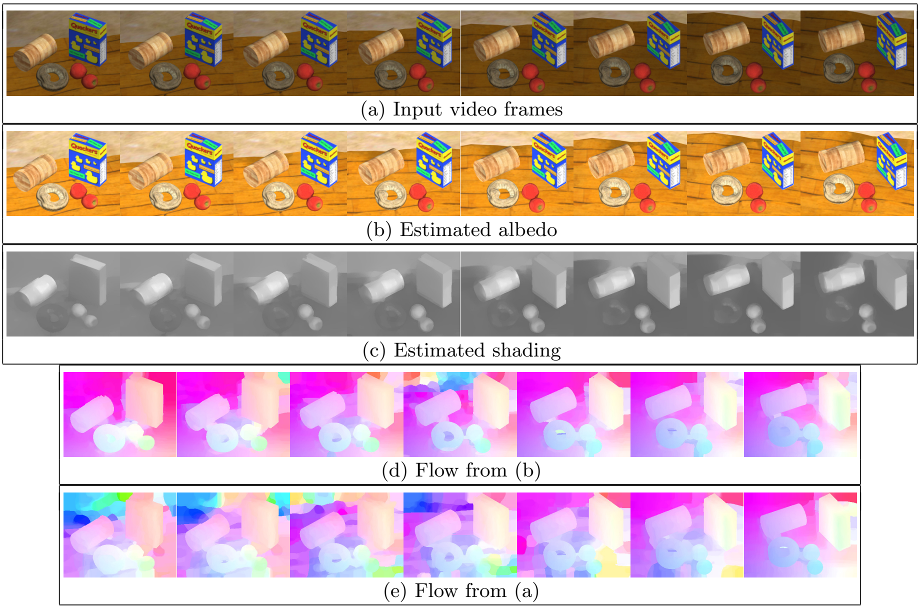
Flow from IV albedo

Flow from original images

Flow from CRET albedo

Flow from GS albedo

Fig. 33. Flow from IV (our method) albedo, original images, CRET albedo, and GS albedo as coherence visualization. The albedo sequence estimated by the previous methods is less coherent than that of ours. The input video is already quite consistent, but our albedo is even more consistent, especially around the top of the cube.



(a) Input video frames

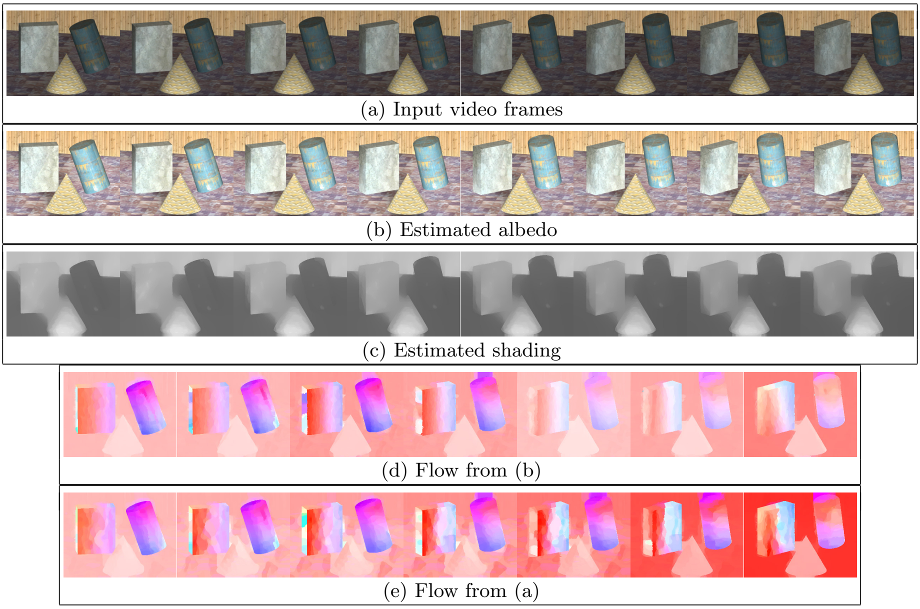
(b) Estimated albedo

(c) Estimated shading

(d) Flow from (b)

(e) Flow from (a)

Fig. 34. Modified synthetic example in Fig. 7 by adding a specular component to some materials. (a) Input video (8 frames; 320 × 240). (b) Albedo estimated by our method (IV). (c) Shading estimated by IV. (d) Flow from (b) as coherence visualization. (e) Flow from (a) as coherence visualization.



(a) Input video frames

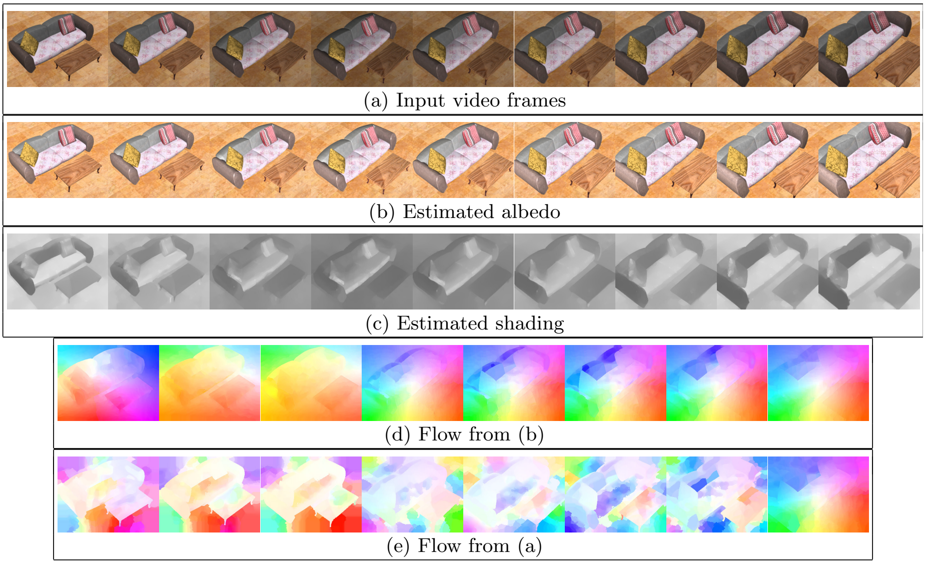
(b) Estimated albedo

(c) Estimated shading

(d) Flow from (b)

(e) Flow from (a)

Fig. 35. Modfied synthetic example in Fig. 12 by adding a specular component to some materials. (a) Input video (8 frames; 320 × 240). (b) Albedo estimated by our method (IV). (c) Shading estimated by IV. (d) Flow from (b) as coherence visualization. (e) Flow from (a) as coherence visualization.



(a) Input video frames

(b) Estimated albedo

(c) Estimated shading

(d) Flow from (b)

(e) Flow from (a)

Fig. 36. Modified synthetic example in Fig. 17 by adding a specular component to some materials. (a) Input video (9 frames; 320 × 240). (b) Albedo estimated by our method (IV). (c) Shading estimated by IV. (d) Flow from (b) as coherence visualization. (e) Flow from (a) as coherence visualization.

References

1. Baker, S., Scharstein, D., Lewis, J.P., Roth, S., Black, M.J., Szeliski, R.: A database and evaluation methodology for optical flow. International Journal of Computer Vision (IJCV) 92(1), 1–31 (2011)
2. Gehler, P., Rother, C., Kiefel, M., Zhang, L., Schölkopf, B.: Recovering intrinsic images with a global sparsity prior on reflectance. In: Shawe-Taylor, J., Zemel, R.S., Bartlett, P.L., Pereira, F.C.N., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems (NIPS). pp. 765–773 (2011)
3. Grosse, R., Johnson, M.K., Adelson, E.H., Freeman, W.T.: Ground-truth dataset and baseline evaluations for intrinsic image algorithms. In: Proc. IEEE International Conference on Computer Vision (ICCV). pp. 2335–2342 (2009)
4. Mac Aodha, O., Humayun, A., Pollefeys, M., Brostow, G.J.: Learning a confidence measure for optical flow. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 35(5), 1107–1120 (2013)
5. Sun, D., Roth, S., Black, M.J.: A quantitative analysis of current practices in optical flow estimation and the principles behind them. International Journal of Computer Vision (IJCV) 106(2), 115–137 (2014)