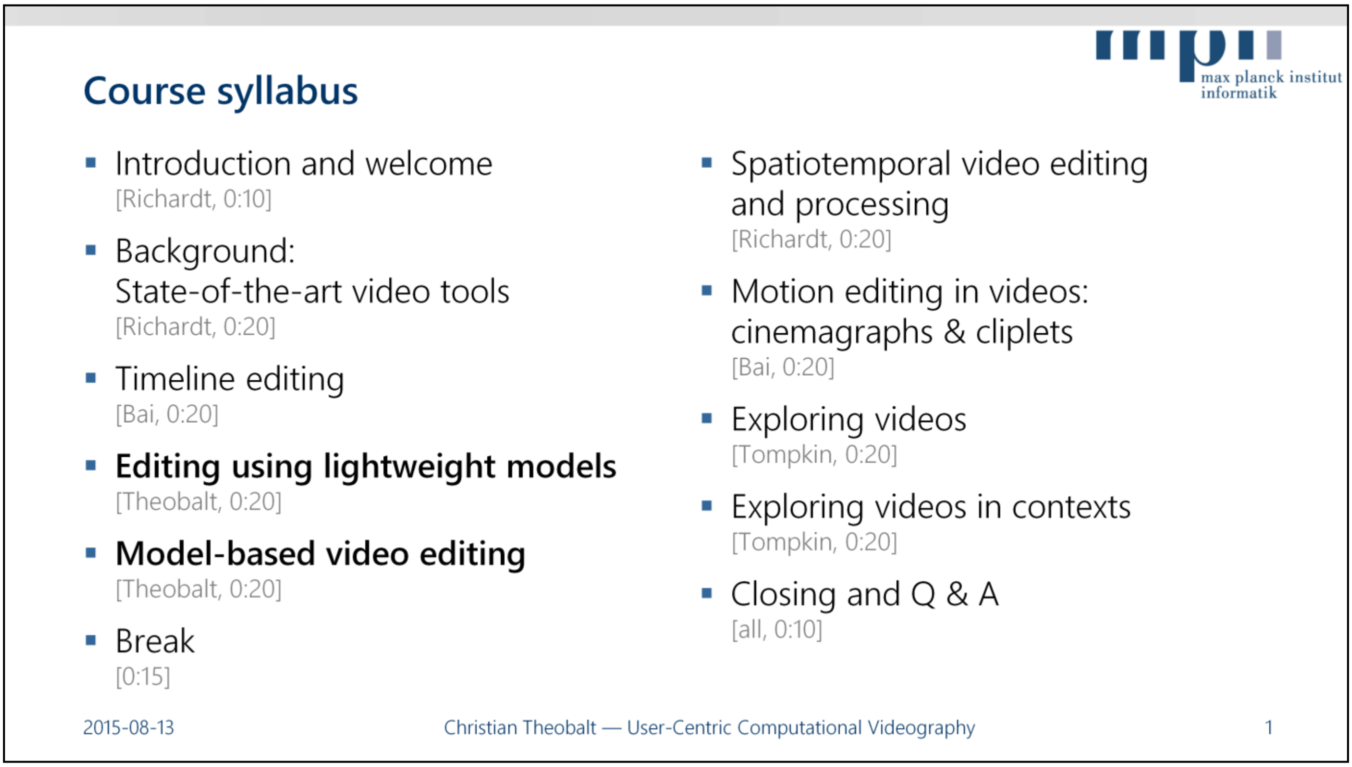
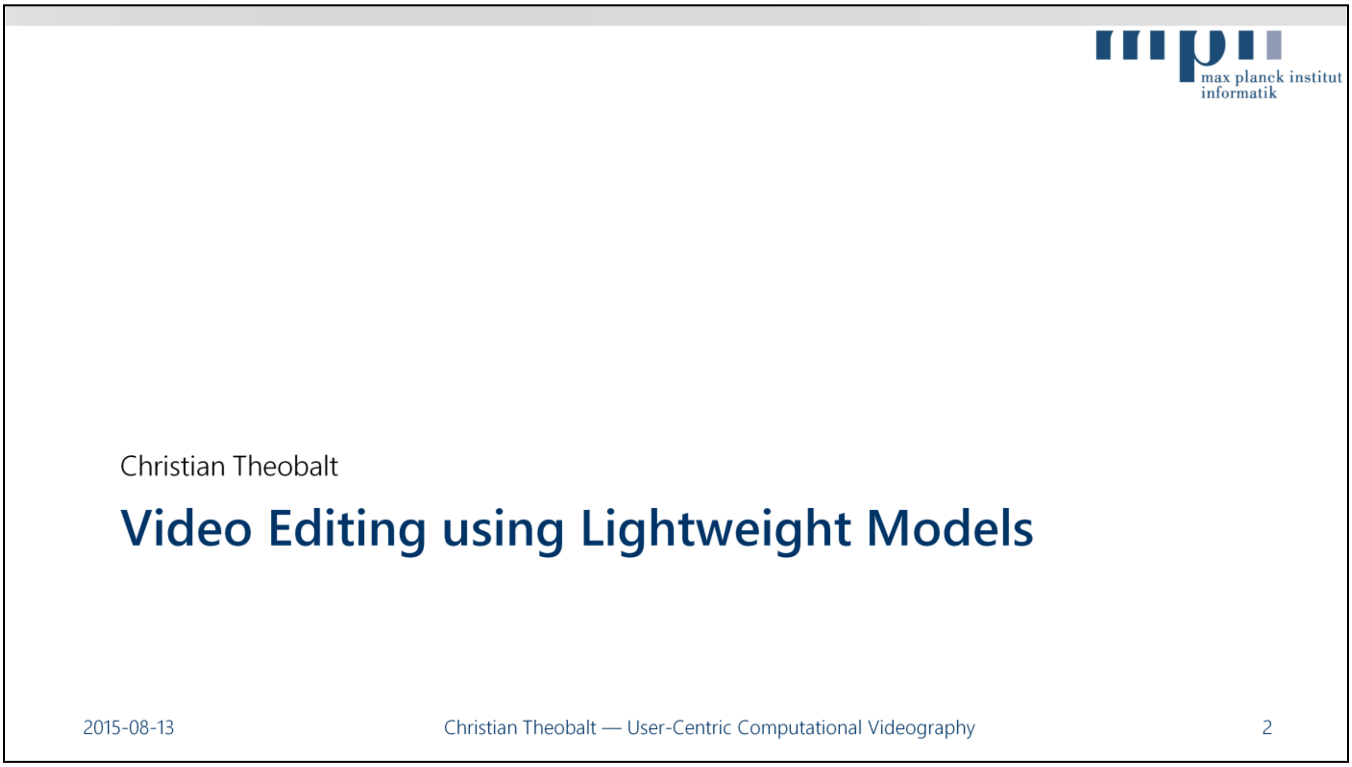


Course syllabus

- Introduction and welcome
  [Richardt, 0:10]
- Background:
  State-of-the-art video tools
  [Richardt, 0:20]
- Timeline editing
  [Bai, 0:20]
- **Editing using lightweight models**
  [Theobalt, 0:20]
- **Model-based video editing**
  [Theobalt, 0:20]
- Break
  [0:15]
- Spatiotemporal video editing and processing
  [Richardt, 0:20]
- Motion editing in videos: cinemagraphs & cliplets
  [Bai, 0:20]
- Exploring videos
  [Tompkin, 0:20]
- Exploring videos in contexts
  [Tompkin, 0:20]
- Closing and Q & A
  [all, 0:10]



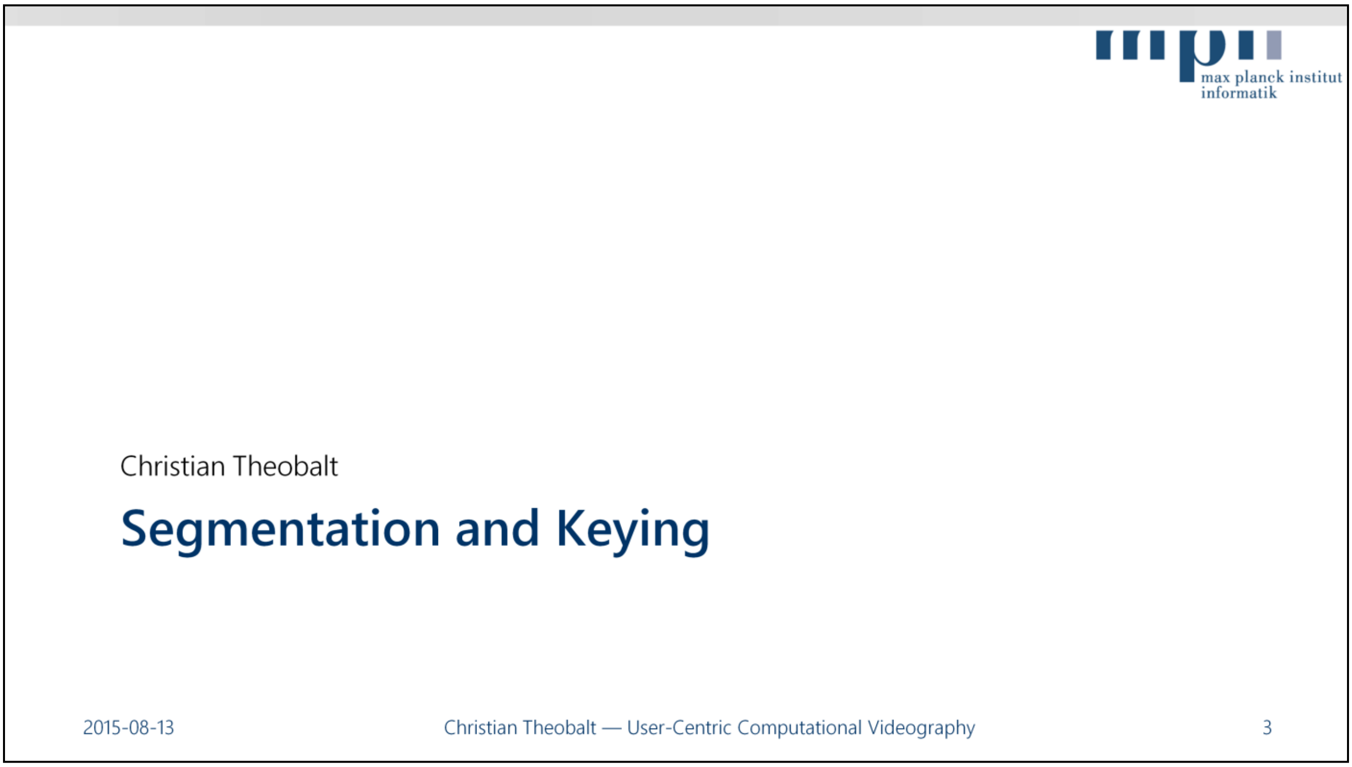
max planck institut informatik

Christian Theobalt

Video Editing using Lightweight Models

2015-08-13 Christian Theobalt — User-Centric Computational Videography 2

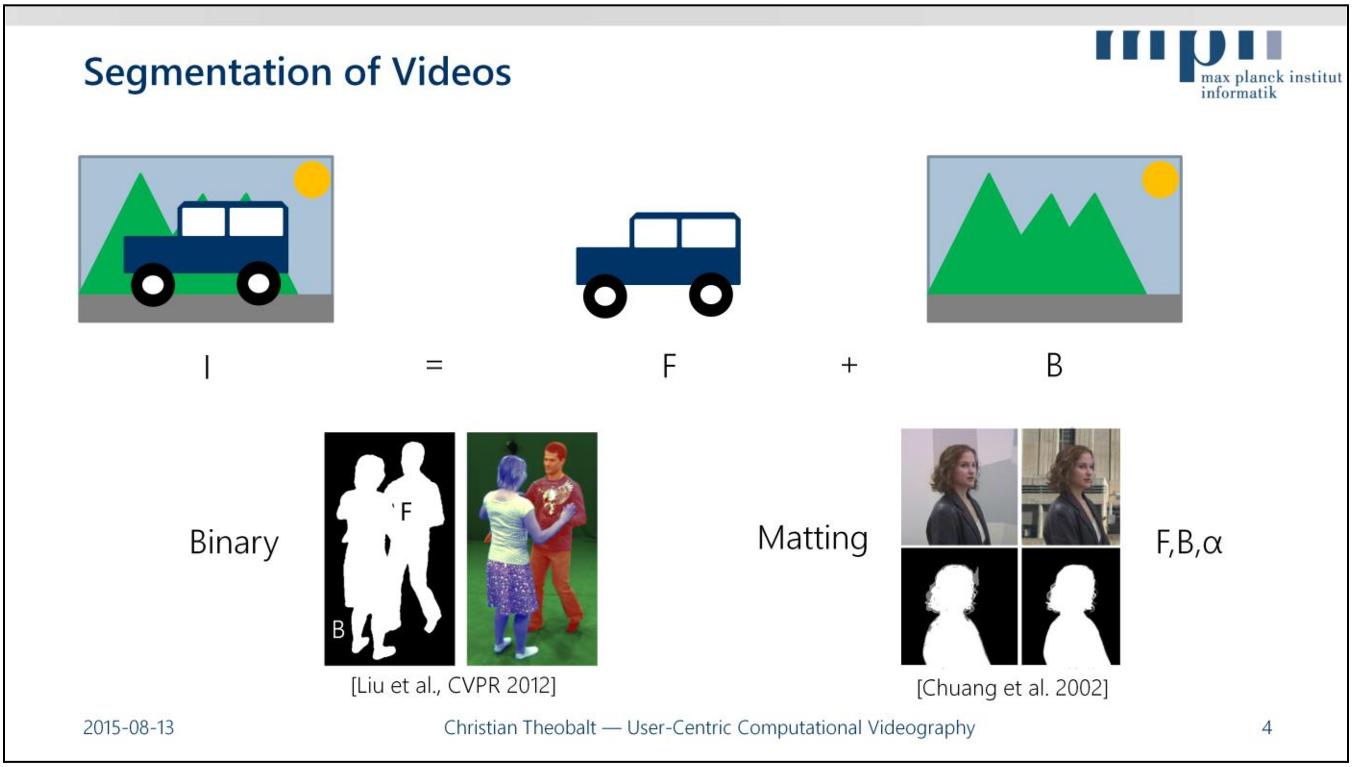
In the following, we will review methods that enable advanced user-centric video editing operations without assuming the availability of a very strong (e.g. shape or appearance) model of the 3D scene that is recorded. These lightweight methods are not approaches that make no prior assumption about the structure of videos, but they make a lot weaker assumptions than having a full 3D reconstruction registered to every video, as we will see later in this part of the talk.



Christian Theobalt

Segmentation and Keying

In the following, we briefly review an important component of many advanced video processing tasks. For many higher level edits, such as replacing an object in video, spatially relocating an object in video, or composting regions from different videos in to one, it is important to have a good segmentation mask of the object. As with many other aspects of video processing that were discussed in the course before, it is important that this segmentation is temporally coherent. So naively applying single image-based segmentation approaches to all frames of a video will generally not serve the purpose, and so dedicated video methods were developed.



Segmentation of Videos

[Liu et al., CVPR 2012]

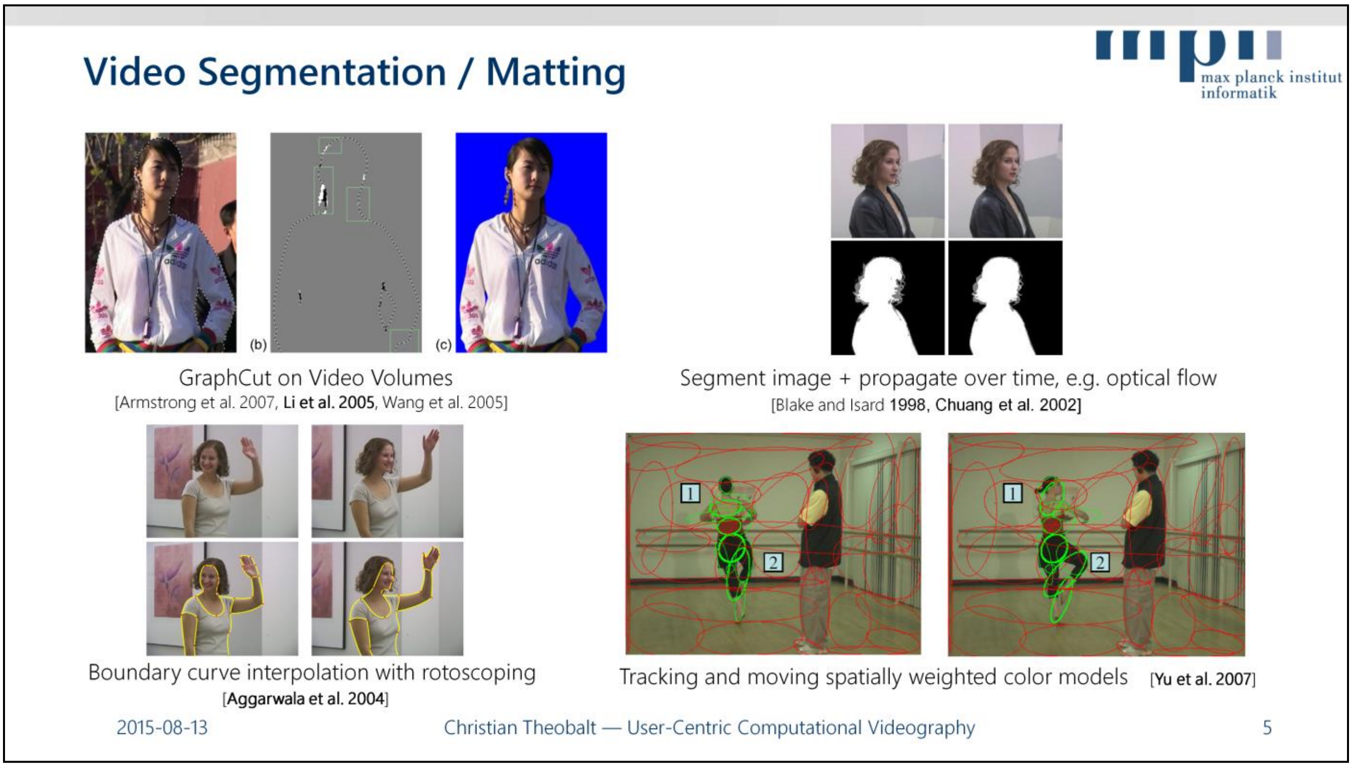
[Chuang et al. 2002]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 4

Let’s briefly define the task we want to solve. The goal is to create a mask for a foreground object **F**, such as the car here, so that it can be separated from the background of a frame **B** .

In the simplest case, the problem can be considered as a binary labeling problem where every pixel in the image is assigned a unique foreground or background label. For many practical applications, however, binary segmentation is not sufficient. Binary masks tend to lead to unwanted artifacts at object boundaries, in particular for scenes where parts of the foreground and background are mixed at a pixel.

Examples are very fine structures of the foreground, such as hair. In order to get a useful segmentation result in this case, one typically formulates a matting problem. In matting, one solves for unique foreground and background labels in regions where one is certain to only see one of the two. In regions where both can mix on a pixel level, such as object boundaries, one defines a so-called trimap area in which one solves, at every pixel, for the foreground and background appearance, but also for the mixing of both, described by an alpha value. The resulting soft segmentation masks are better suited for many applications, such as compositing.



Video Segmentation / Matting

GraphCut on Video Volumes
[Armstrong et al. 2007, Li et al. 2005, Wang et al. 2005]

Segment image + propagate over time, e.g. optical flow
[Blake and Isard 1998, Chuang et al. 2002]

Boundary curve interpolation with rotoscoping
[Aggarwala et al. 2004]

Tracking and moving spatially weighted color models [Yu et al. 2007]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 5

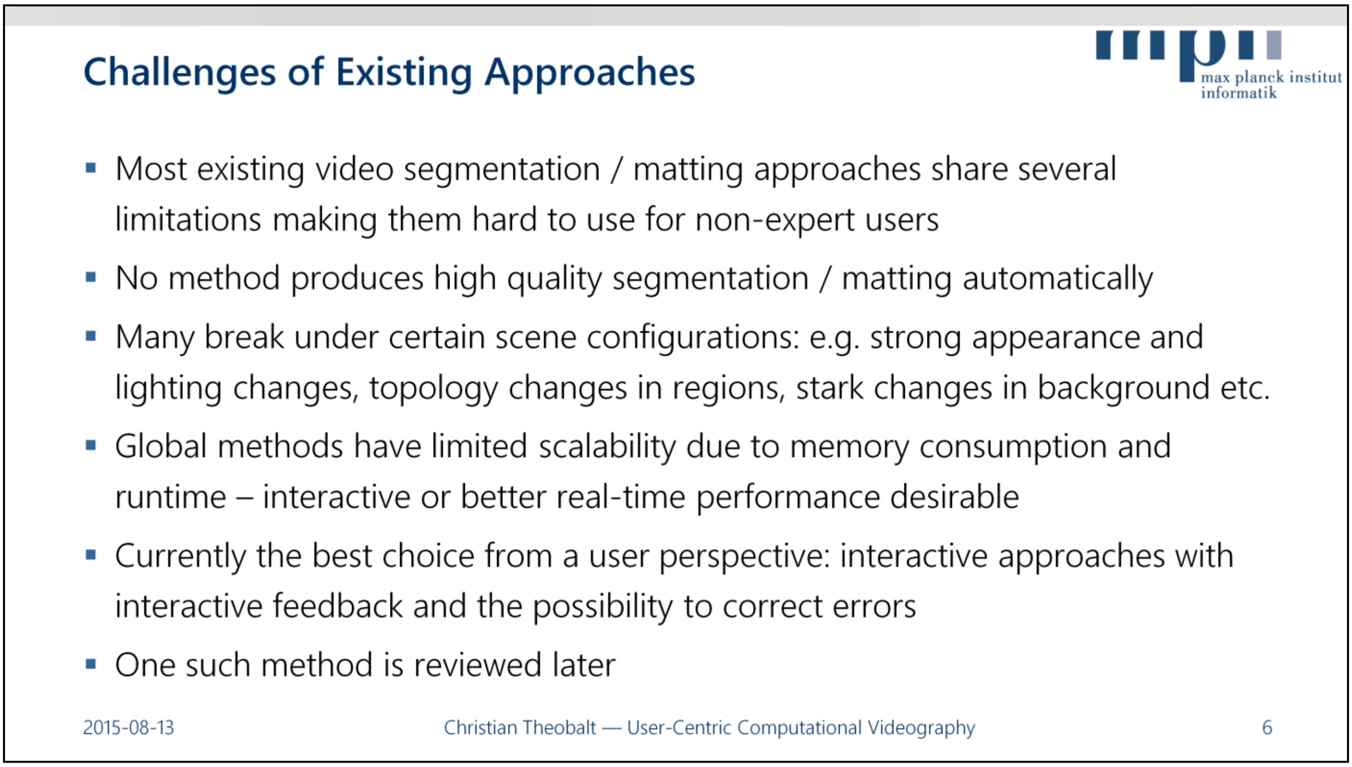
There is a large body of prior work on image segmentation and matting [Wang et al. 2007] and recent works have reached a level of maturity such that they can be integrated into commercial products. The need for spatio-temporal coherence and stability in video segmentation and matting makes it a harder problem.

Even though not yet comparable to the scope of image-based works, there is a fair amount of literature on video segmentation and matting [Varnousfaderani et al. 2013], an extensive review is beyond the scope of this course, but we review the main categories of approaches. Here are examples of widely used algorithmic categories:

1) Some methods extends the graph-cut-based image segmentation to video volumes. This means global optimization of foreground / background labels in video frames is solved. They often require manual correction of errors in the globally optimized solution.
2) Other approaches combine per image segmentation / matting with propagation of information over time, e.g. using optical flow.
3) Many methods rely on color-based appearance models and track and move spatially weighted color models.
4) Another category of methods interpolates boundary curves, such as splines, using a rotoscoping approach.

While having advanced greatly in recent years, none of these produces satisfactory segmentation or matting results in all possible cases, in particular not fully-automatically. Often times, manual user intervention is needed, but not all approaches enable that. Common limitations of global approaches are scalability problems due to memory limitations, as well as the inability to capture foreground and background appearance appropriately in a global model. A variety of other limitations exist that are summarized on the next slide.

Solving the involved optimization problems is also more challenging and often time consuming in global approaches, which may prevent interactive methods in the first place. These problems in the single image case are amplified for videos. The problem complexities are even larger, and since user correction will often be needed to obtain good results, runtime performance is even more of an issue.

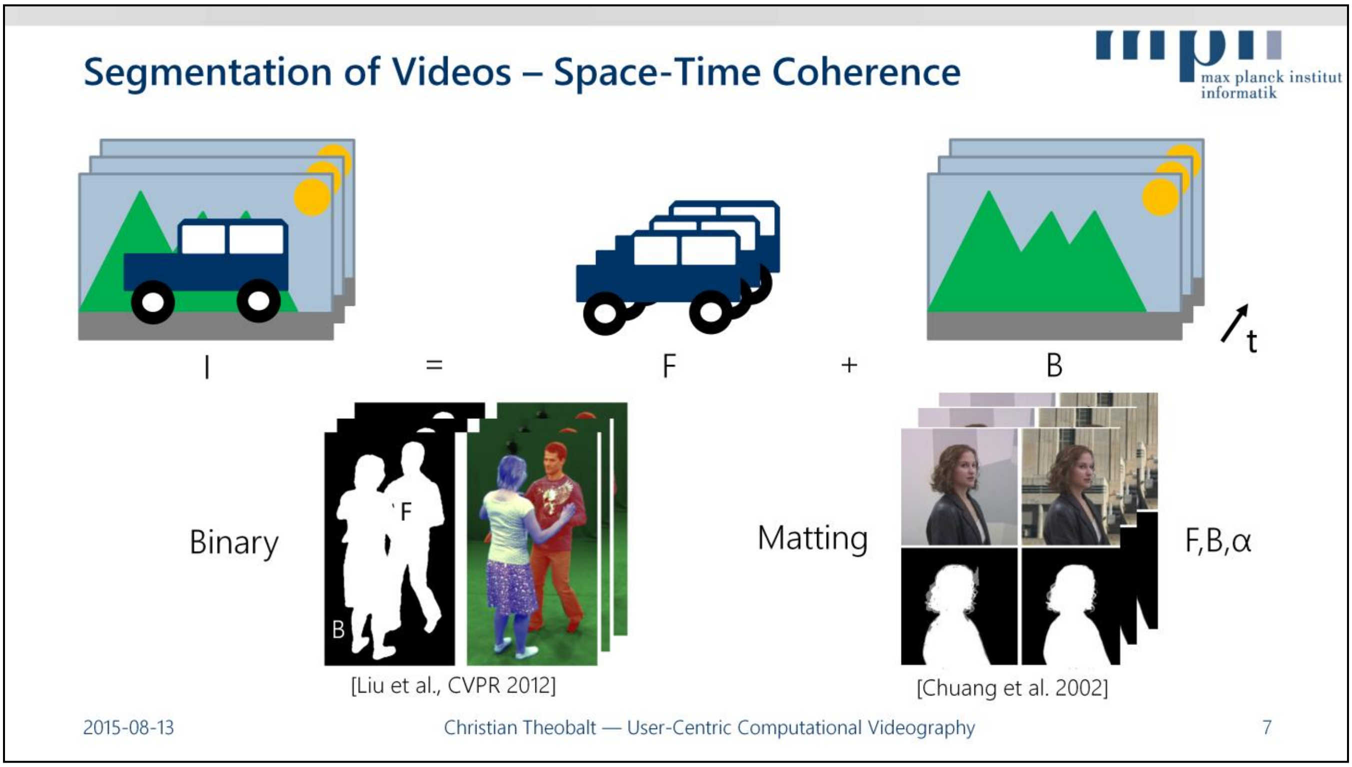


Challenges of Existing Approaches

- Most existing video segmentation / matting approaches share several limitations making them hard to use for non-expert users
- No method produces high quality segmentation / matting automatically
- Many break under certain scene configurations: e.g. strong appearance and lighting changes, topology changes in regions, stark changes in background etc.
- Global methods have limited scalability due to memory consumption and runtime – interactive or better real-time performance desirable
- Currently the best choice from a user perspective: interactive approaches with interactive feedback and the possibility to correct errors
- One such method is reviewed later

2015-08-13 Christian Theobalt — User-Centric Computational Videography 6

This slide lists again some of the main challenges that existing video segmentation and matting approaches are facing, and explains why, from a user perspective, interactive approaches are a good choice.

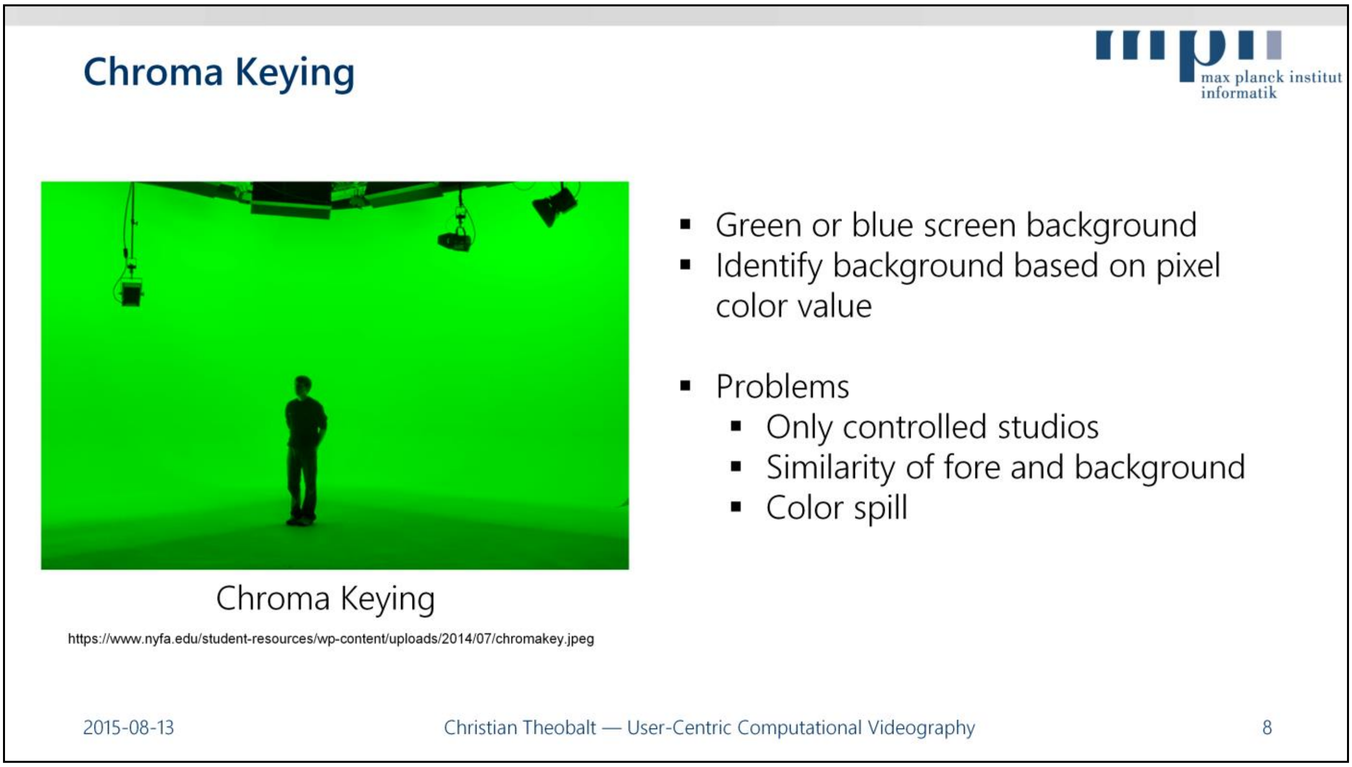


Segmentation of Videos – Space-Time Coherence

[Liu et al., CVPR 2012]

[Chuang et al. 2002]

As stated before, naively applying one of the many image-based segmentation or matting approaches to each frame of video individually is most likely not going to solve the problem in a temporally stable way. Specific video segmentation approaches have to live up to this additional requirement of temporal coherence.



Chroma Keying

- Green or blue screen background
- Identify background based on pixel color value

- Problems
  - Only controlled studios
  - Similarity of fore and background
  - Color spill

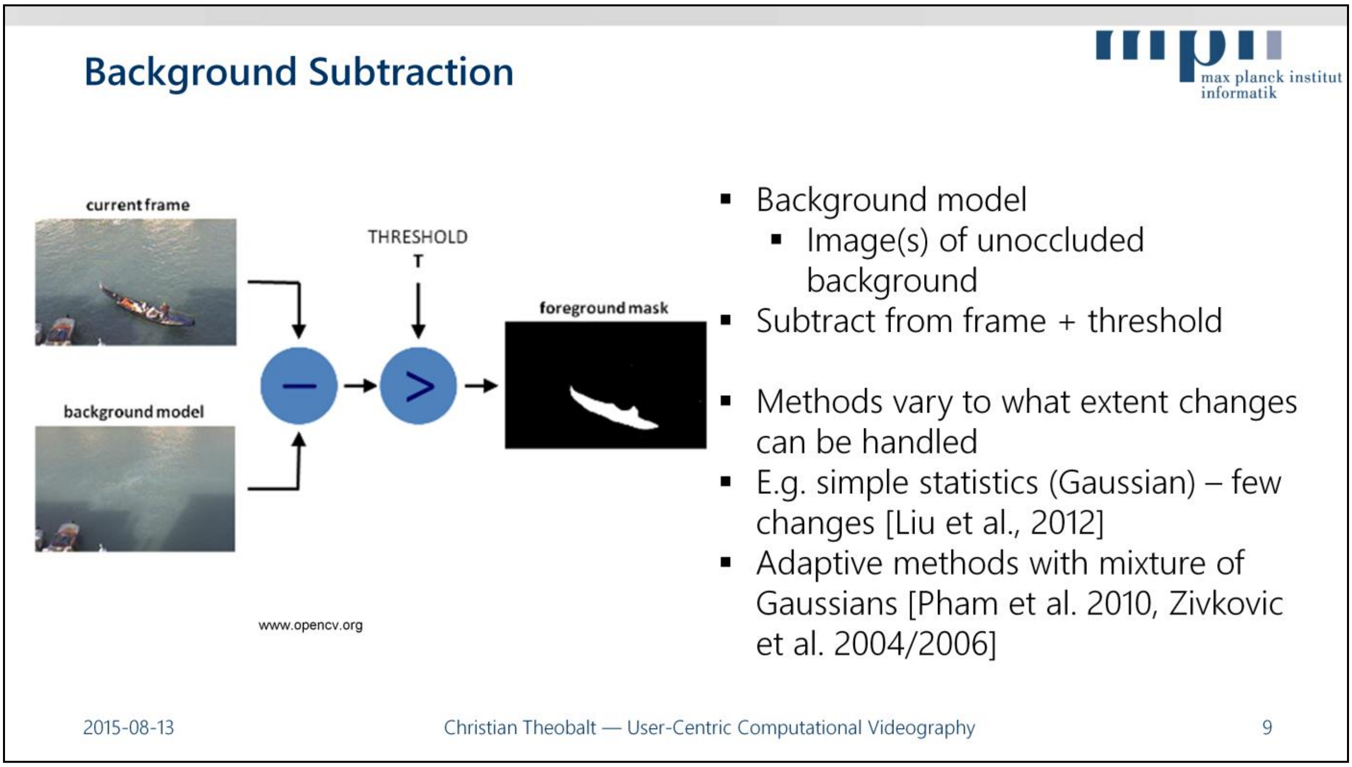
Chroma Keying

https://www.nyfa.edu/student-resources/wp-content/uploads/2014/07/chromakey.jpeg

2015-08-13 Christian Theobalt — User-Centric Computational Videography 8

One way of segmenting or matting the foreground from the background is to engineer the environment. This is not possible in general scenes, but feasible under controlled studio conditions.

A widely used approach in many studio productions is chroma keying. The idea is to record the foreground in front of a backdrop of specific color, typically green or blue. With that engineered background, the foreground can be separated from the background by finding background colored pixels. Of course, this is only feasible under fully controlled conditions, and also has several failure conditions in practice. In particular, segmentation may fail if foreground and background colors overlap. Also, the mono-colored backdrop often leads to unwanted color spill (interrecflection) onto the foreground.



Background Subtraction

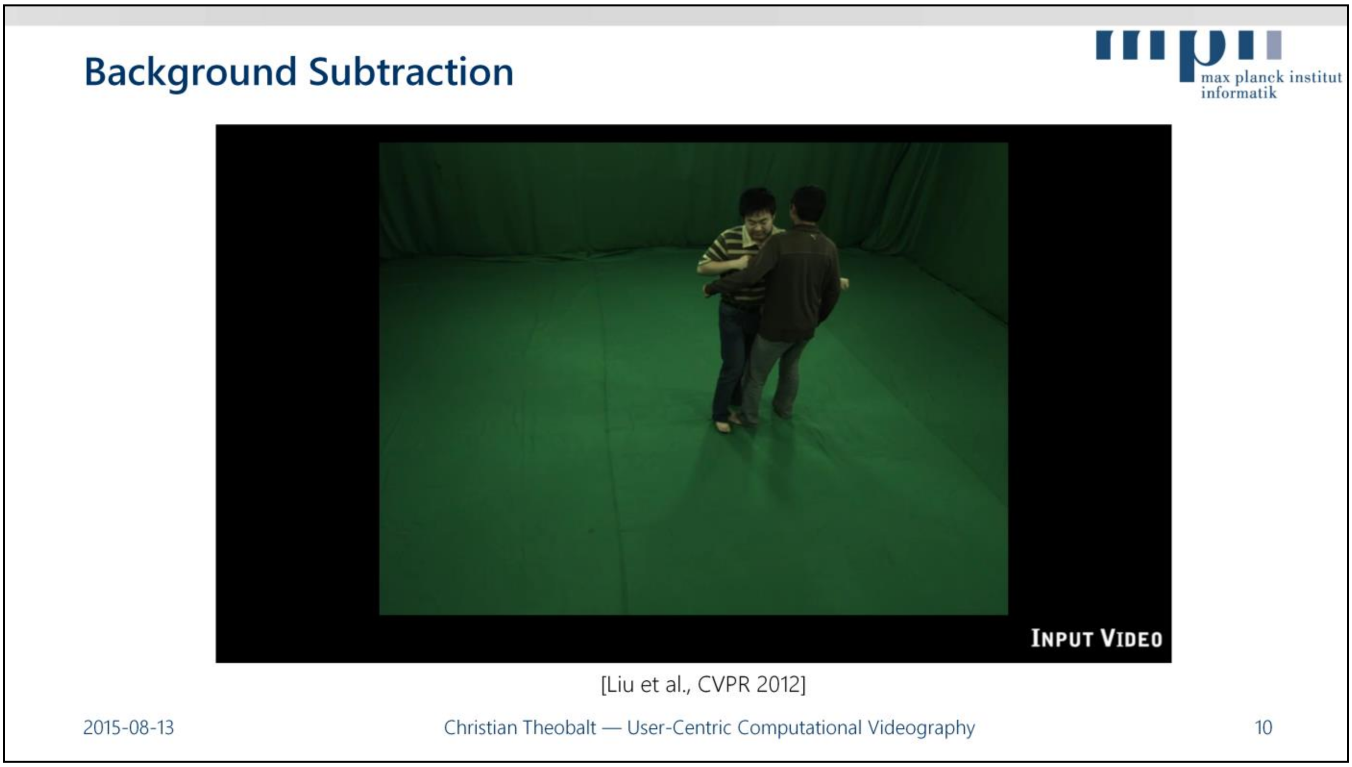
max planck institut informatik

- Background model
  - Image(s) of unoccluded background
- Subtract from frame + threshold

- Methods vary to what extent changes can be handled
- E.g. simple statistics (Gaussian) – few changes [Liu et al., 2012]
- Adaptive methods with mixture of Gaussians [Pham et al. 2010, Zivkovic et al. 2004/2006]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 9

Chroma keying is an instance of a slightly more general class of approaches, namely background subtraction methods. Background subtraction is based on a simple idea: if one can record a clean background image one can subtract the background from the foreground (on a per-pixel or per patch level) by thresholding the difference between an image and the backdrop. Existing background subtraction methods differ in several ways: while simple approaches can only work with static cameras and no appearance changes in the background, more advanced approaches can handle more variations, such as slight camera motion or appearance and lighting variation over time. Simple approaches thus builds simple per pixel statistics of the background, such as a Gaussian color model [Li et al. 2012]. More advanced methods learn adaptive background statistics using, for instance, adaptive Gaussian mixture models, e.g. [Pham et al. 2010, Zivkovic et al. 2004/2006].

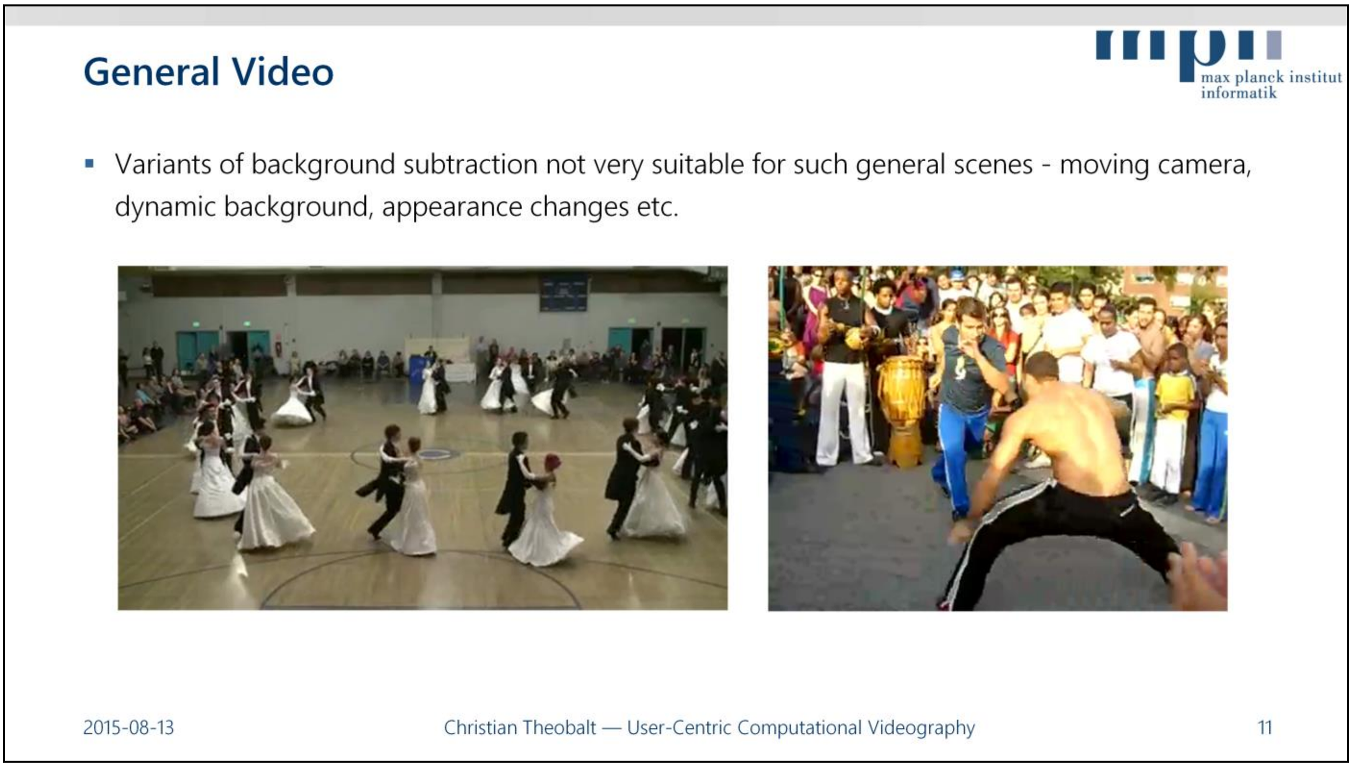


Background Subtraction

[Liu et al., CVPR 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 10

The video above shows an example of a multi-view performance capture method that uses as input a set of multi-view video sequences recorded in a studio. Foreground silhouettes for each camera view are extracted by means of a background subtraction approach.

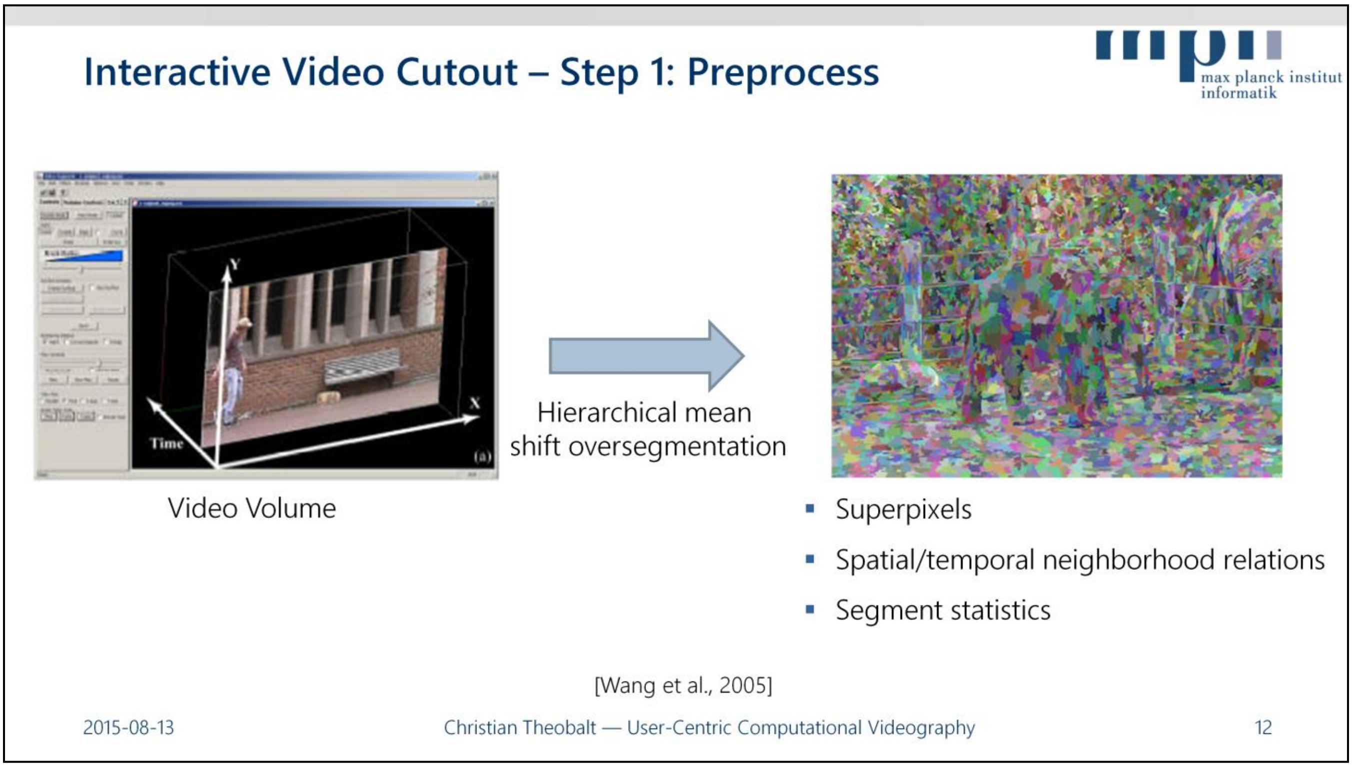


General Video

- Variants of background subtraction not very suitable for such general scenes - moving camera, dynamic background, appearance changes etc.

2015-08-13 Christian Theobalt — User-Centric Computational Videography 11

Chroma keying in particular, and background subtraction approaches in general are challenged by the complexity of many real world video sequences. Real world videos often contain very dynamic and starkly changing backgrounds, exhibit very notable camera motion, and exhibit strong changes in appearance. These variations are usually not captured very well by the models of foreground and background used in many of the aforementioned approaches.



Interactive Video Cutout – Step 1: Preprocess

Video Volume

Hierarchical mean shift oversegmentation

- Superpixels
- Spatial/temporal neighborhood relations
- Segment statistics

[Wang et al., 2005]

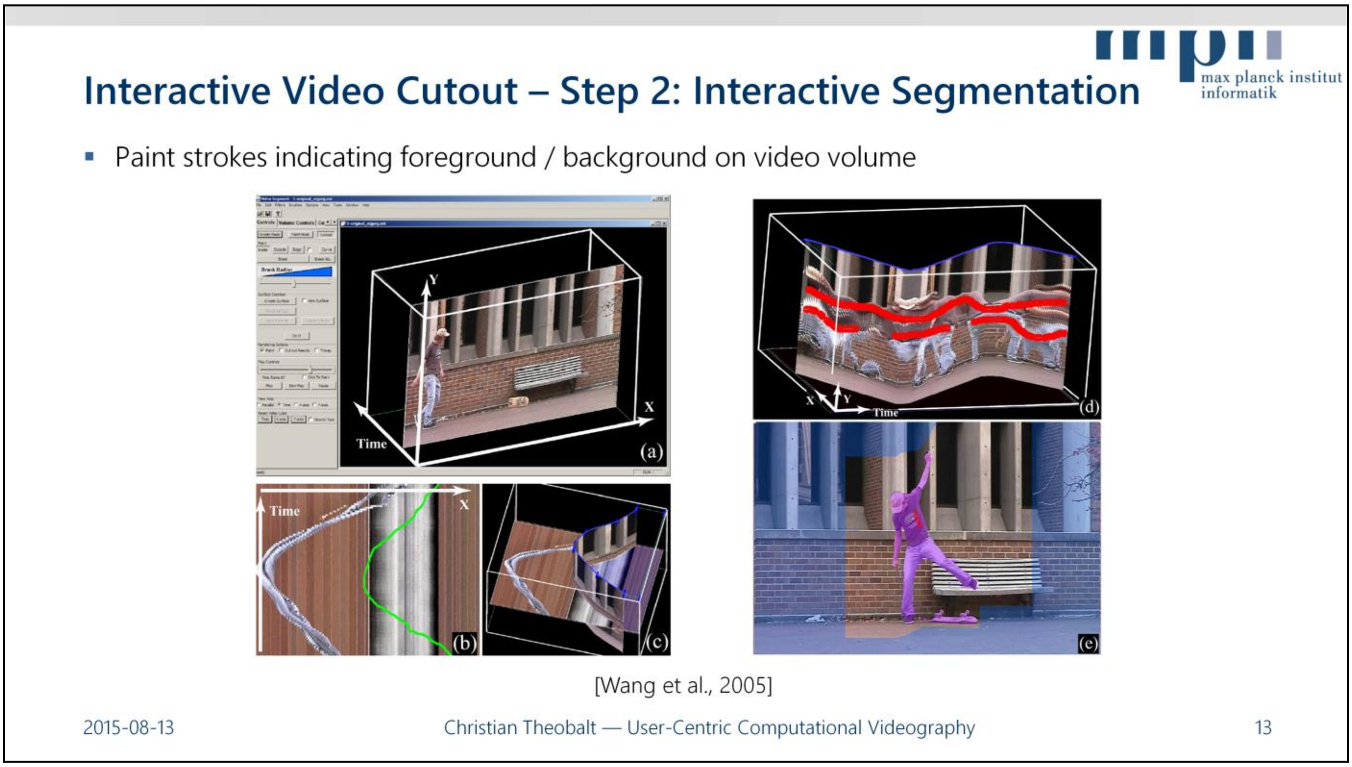
2015-08-13 Christian Theobalt — User-Centric Computational Videography 12

In the following slides, we review an example approach from the literature, the Interactive Video Cutout algorithm by Wang et al., that enables temporally coherent video segmentation and matting of foreground regions with user interaction. The method is designed in such a way that the involved computations can be performed efficiently, enabling video segmentation with interactive user feedback and refinement. It is also applicable to general scenes. In the following, the main steps and core ideas are briefly reviewed.

As mentioned before, the complexity of video segmentation is amplified by having to solve for segmentation results in a temporally coherent way. Phrasing video segmentation as a global problem on the entire video volume (temporal stacking of subsequent video frames into a common data structure, visualized in figure above) is computationally prohibitive. Therefore, a pre-processing step in Wang et al.'s method performs a hierarchical oversegmentation.

First, each frame in a video volume is segmented separately based on appearance information, i.e. pixels are grouped into 2D superpixels – larger coherent image regions of visual similarity. Subsequently 2D superpixels are clustered into 3D spatio-temporal regions based on spatial neighborhood and appearance similarity. The oversegmentation is hierarchical and unique, i.e. each pixel is a member of exactly one 2D region, and each 2D region is a member of exactly one 3D region. Certain appearance statistics for regions (henceforth also termed segments) are also computed for later use.

The oversegmentation result can be visualized in a tree (see later slide).



max planck institut informatik

Interactive Video Cutout – Step 2: Interactive Segmentation

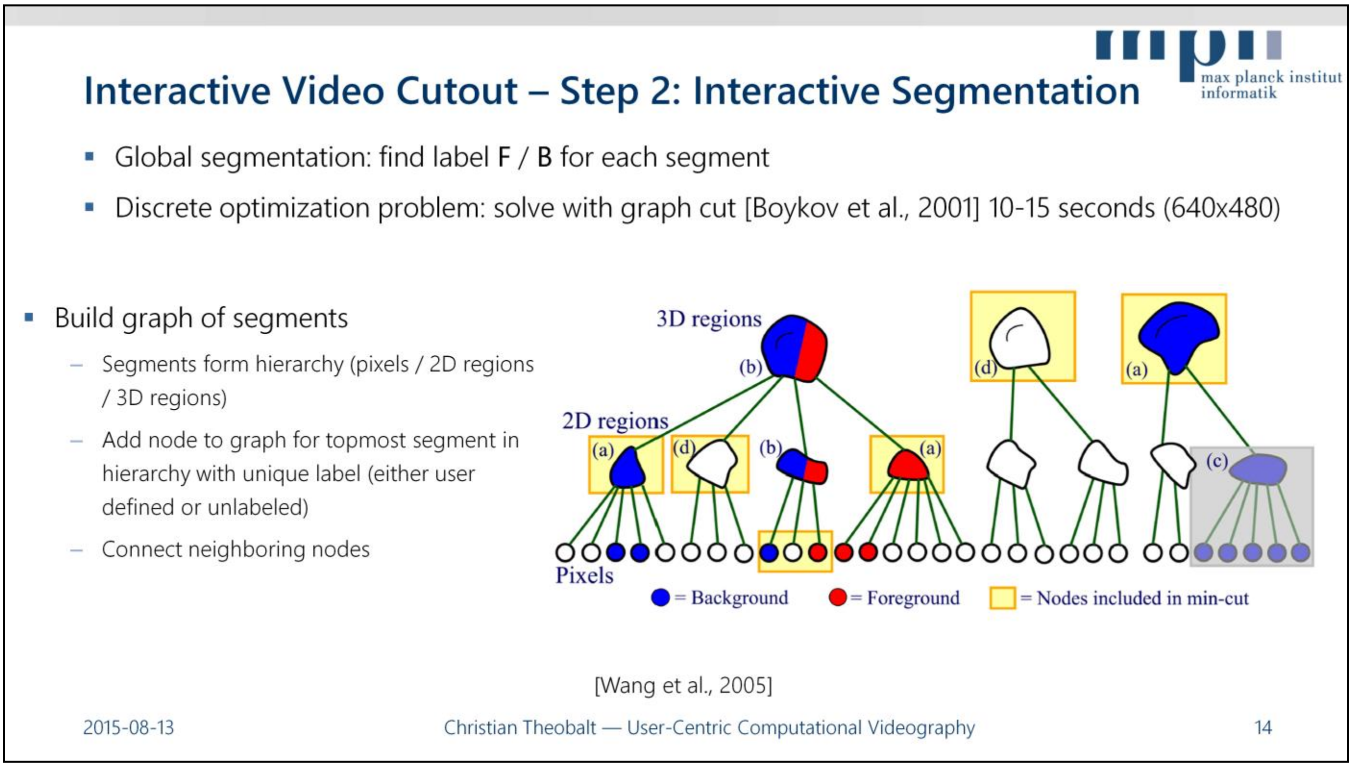
- Paint strokes indicating foreground / background on video volume

[Wang et al., 2005]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 13

After preprocessing is completed, the user is asked to provide rough scribbles on the video volume on regions that he considers to be foreground and on regions that he considers to be background. To this end, a 3D painting interface was designed in which the user can rotate and interact with the video volume, and provide strokes by painting on the respective spatio-temporal sub-dimensions.

The Video Cutout method now enables the user to alternate between painting of strokes and computation of a segmentation result. This interactive scheme enables the refinement of a segmentation from the previous iteration, in case there were errors, by providing additional user input where it is needed.



Interactive Video Cutout – Step 2: Interactive Segmentation

- Global segmentation: find label **F** / **B** for each segment
- Discrete optimization problem: solve with graph cut [Boykov et al., 2001] 10-15 seconds (640x480)

- Build graph of segments
  - Segments form hierarchy (pixels / 2D regions / 3D regions)
  - Add node to graph for topmost segment in hierarchy with unique label (either user defined or unlabeled)
  - Connect neighboring nodes

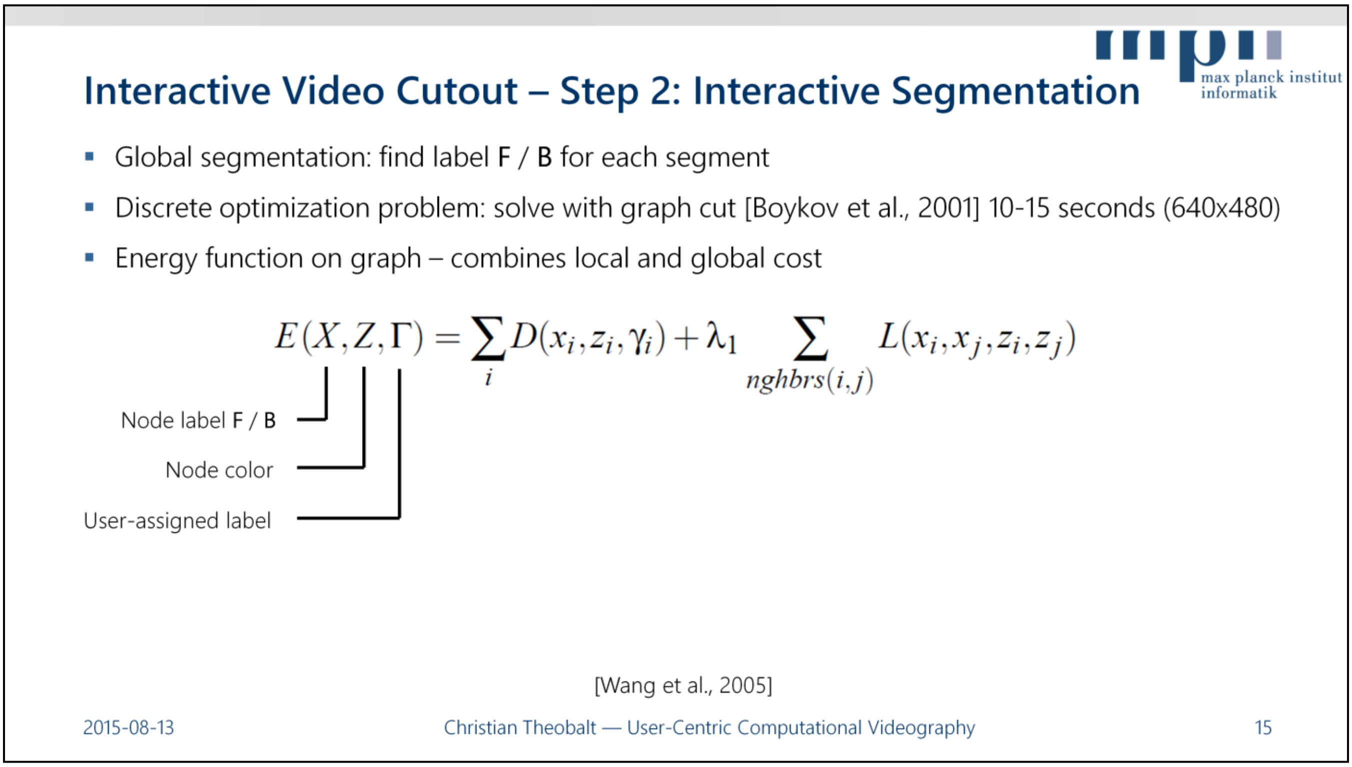
[Wang et al., 2005]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 14

Given the user scribbles, the algorithm proceeds by computing a segmentation on the entire video volume. This is formulated as a global optimization problem that assigns to each segment in the video volume a unique foreground (**F**) or background (**B**) label. The optimization is performed as a graph-cut (min-cut) computation [Boykov et al. 2001] on a specifically structured neighborhood graph.

Let us thus first see how the graph on which the optimization problem is solved is constructed. The nodes of the graph are segments in the video volume. Remember that from the preprocessing step there is a hierarchical decomposition of the video volume into segments, where a segment can either be a pixel (lowest level in the hierarchy tree visualized above), a 2D region (superpixel), or a 3D region. The goal is to decide what segments of the hierarchy are to be added to the graph. From user strokes, some pixels in the video volume were given foreground (red) or background (blue) labels, unlabeled pixels are shown in white. The idea is to add to the graph the topmost segments in the segment hierarchy that have a unique label, i.e. **F**,**B** or unlabeled. 2D or 3D regions that have pixels with differing labels are not to be added.

The graph is then formed from these nodes by connecting spatio-temporally adjacent segments in the video volume.



Interactive Video Cutout – Step 2: Interactive Segmentation

- Global segmentation: find label **F** / **B** for each segment
- Discrete optimization problem: solve with graph cut [Boykov et al., 2001] 10-15 seconds (640x480)
- Energy function on graph – combines local and global cost

$$E(X,Z,\Gamma) = \sum_i D(x_i,z_i,\gamma_i) + \lambda_1 \sum_{nghbrs(i,j)} L(x_i,x_j,z_i,z_j)$$

Node label **F** / **B**

Node color

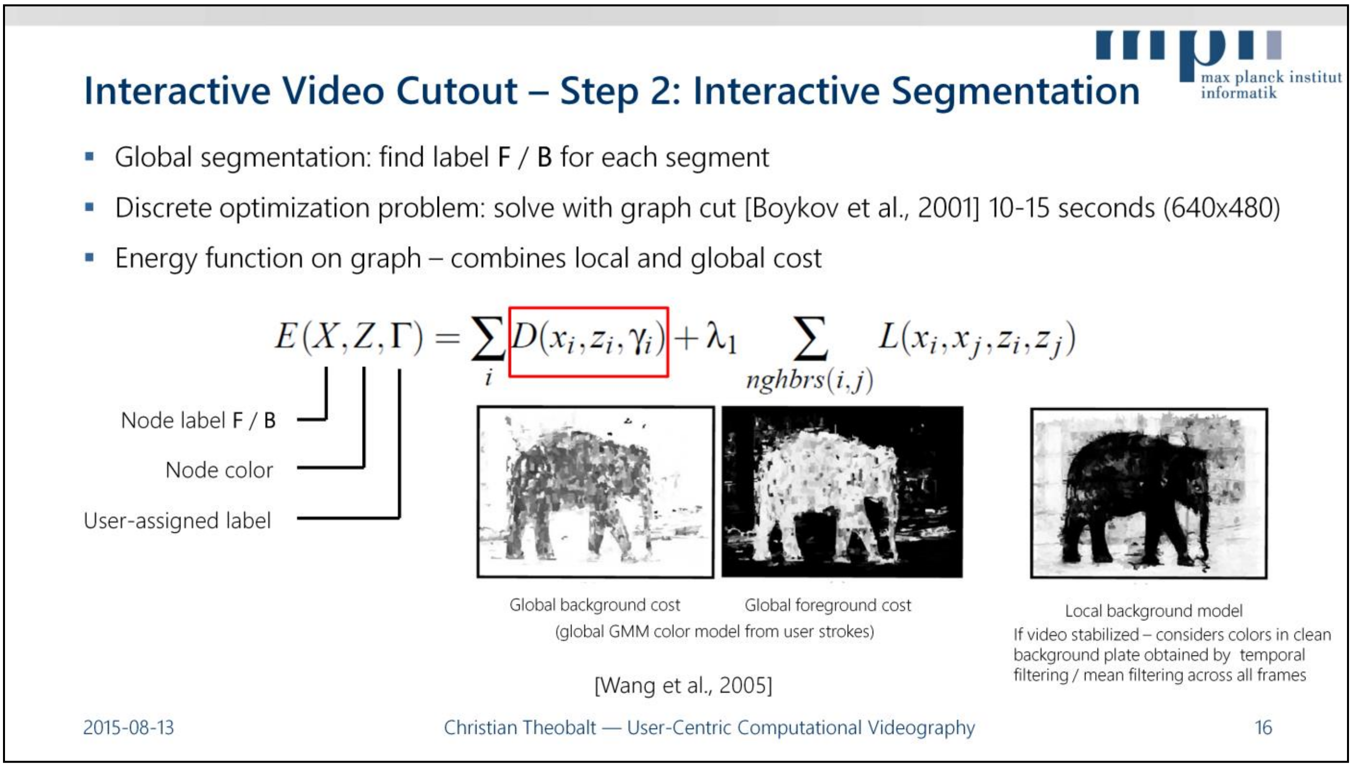
User-assigned label

[Wang et al., 2005]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 15

After we know how the graph is built, let’s now define the energy function on the graph whose minimum defines the desired segmentation, i.e. a foreground (**F**) or background (**B**) label for each node, **X**.

The energy takes as additional input a color model for each node, **Z**, as well as the user-assigned labels from the painted strokes, **Γ**. The first term defines a data term (unary potential) on each node, and the second term is a pairwise spatial regularizer of the global segmentation result. In the following, unary and pairwise terms are discussed in more detail.



Interactive Video Cutout – Step 2: Interactive Segmentation

- Global segmentation: find label **F** / **B** for each segment
- Discrete optimization problem: solve with graph cut [Boykov et al., 2001] 10-15 seconds (640x480)
- Energy function on graph – combines local and global cost

$$E(X,Z,\Gamma) = \sum_i \boxed{D(x_i,z_i,\gamma_i)} + \lambda_1 \sum_{nghbrs(i,j)} L(x_i,x_j,z_i,z_j)$$

Node label **F** / **B**

Node color

User-assigned label

Global background cost

Global foreground cost

(global GMM color model from user strokes)

Local background model

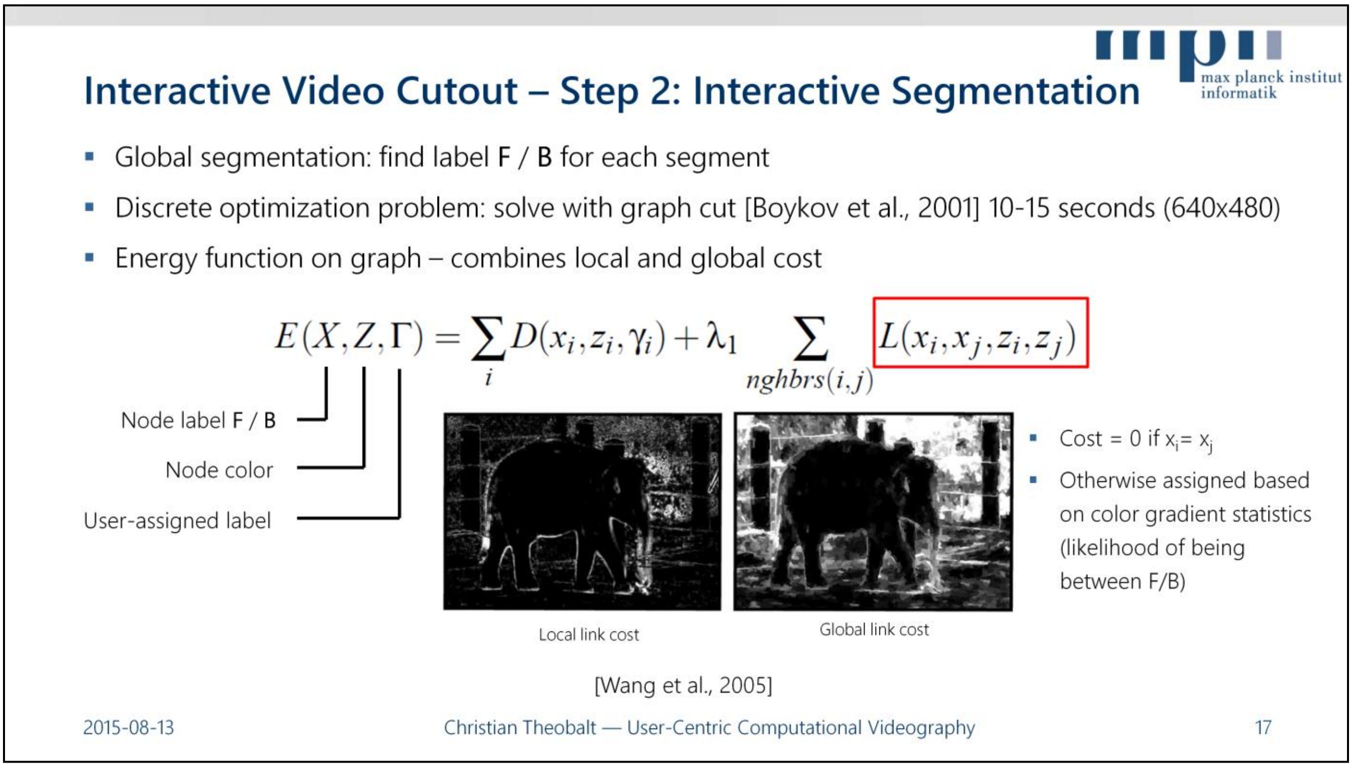
If video stabilized – considers colors in clean background plate obtained by temporal filtering / mean filtering across all frames

[Wang et al., 2005]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 16

The data term has 3 components. First, there are two global terms which describe the likelihood that a segment belongs to the foreground or the background according to a global appearance model of foreground and background that was learned from the regions with provided scribbles. In the above slide, the foreground and background likelihoods of segments in one frame of video are exemplarily visualized (white means high likelihood).

As mentioned earlier, global appearance models are limited in their expressive power as they often fail to properly represent more localized foreground and background statistics in specific regions. Therefore, there is an additional local background model. The assumption behind this model is that the camera is stable or the video was stabilized and thus a hypothesis of a background image can be extracted by spatio-temporal filtering of the video volume, e.g. a mean filter over time on each pixel. From this backdrop, local background statistics can be computed and used to compute an additional background likelihood based on the difference to the local model.



Interactive Video Cutout – Step 2: Interactive Segmentation

- Global segmentation: find label **F** / **B** for each segment
- Discrete optimization problem: solve with graph cut [Boykov et al., 2001] 10-15 seconds (640x480)
- Energy function on graph – combines local and global cost

$$E(X,Z,\Gamma) = \sum_i D(x_i,z_i,\gamma_i) + \lambda_1 \sum_{nghbrs(i,j)} \boxed{L(x_i,x_j,z_i,z_j)}$$

Node label **F** / **B**

Node color

User-assigned label

Local link cost

Global link cost

- Cost = 0 if $x_i = x_j$
- Otherwise assigned based on color gradient statistics (likelihood of being between F/B)

[Wang et al., 2005]

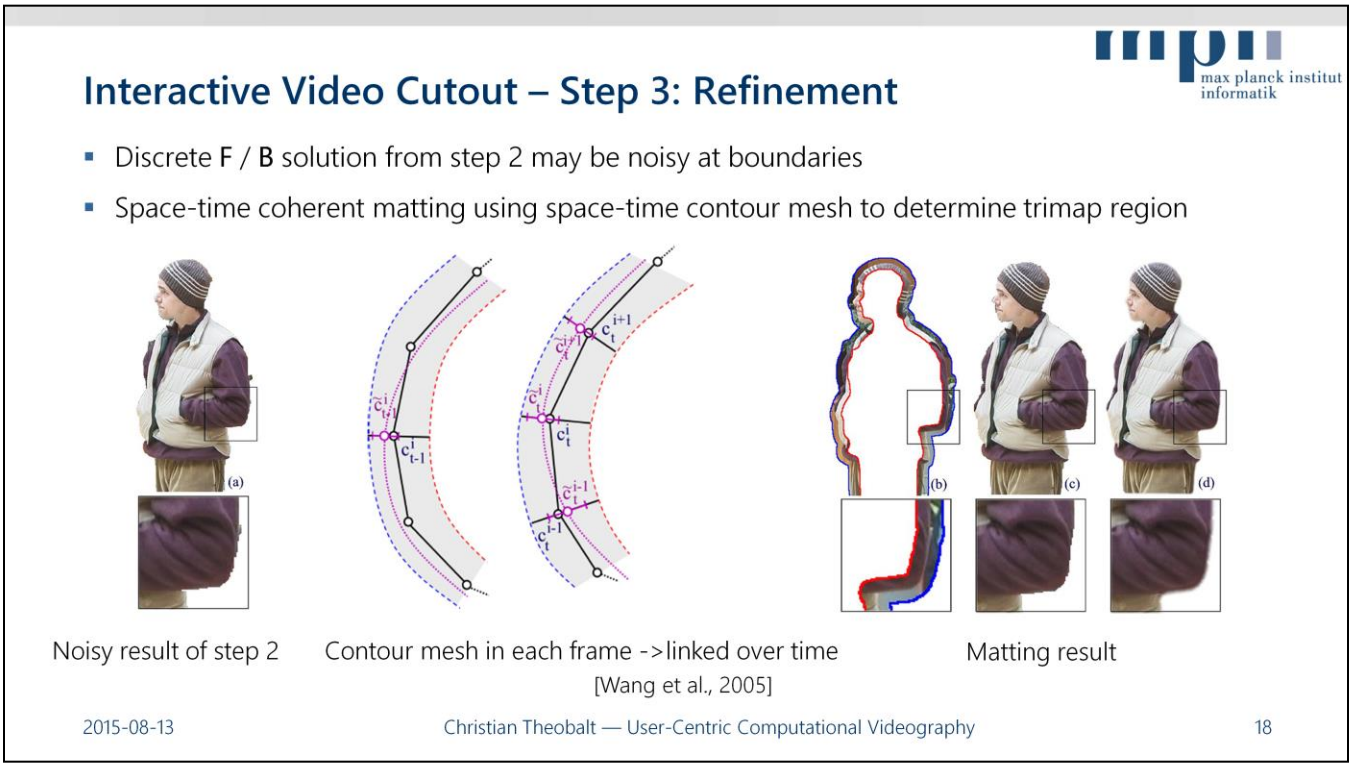
White indicates low cost – i.e. good place to cut a link. The link cost is a regularization term based on the relationship between adjacent nodes that enforces spatially and temporally coherent segmentations. The link cost is 0 if the labels assigned to the neighboring nodes are identical.

As for the data term, the link cost combines evidence from a global and a local term.

The global link cost encodes the following observation: Transitions between foreground and background will usually exhibit higher gradient than the average gradient between any two pixels. This global link cost is phrased as a static exponential function of gradient (color difference) between nodes whose parameters are found from the gradient statistics of the labeled nodes.

In contrast, the local link cost encodes relations based on so-called pixel and link spans: a pixel span is a set of pixels occurring over time at the same location x/y. A link span is a set of links connecting neighboring pixel (or node) spans. The local link cost computes the statistic of gradients in link spans and penalizes strong outliers from that statistics. This shall help to better preserve natural gradients at the interior of textured foreground / background regions, and to differentiat them from true foreground background changes.

None of the individual cost terms is perfect by itself, but in tandem they enable to solve for foreground an background faithfully. The energy is optimized via a min-cut solve on the graph (see Boykov et al.). The entire step 2 of interactive video cutout can be iterated, i.e. new strokes can be added and a new segmentation can be solved via graph cut if the previous result was unsatisfactory.



Interactive Video Cutout – Step 3: Refinement

- Discrete **F** / **B** solution from step 2 may be noisy at boundaries
- Space-time coherent matting using space-time contour mesh to determine trimap region

Noisy result of step 2

Contour mesh in each frame ->linked over time

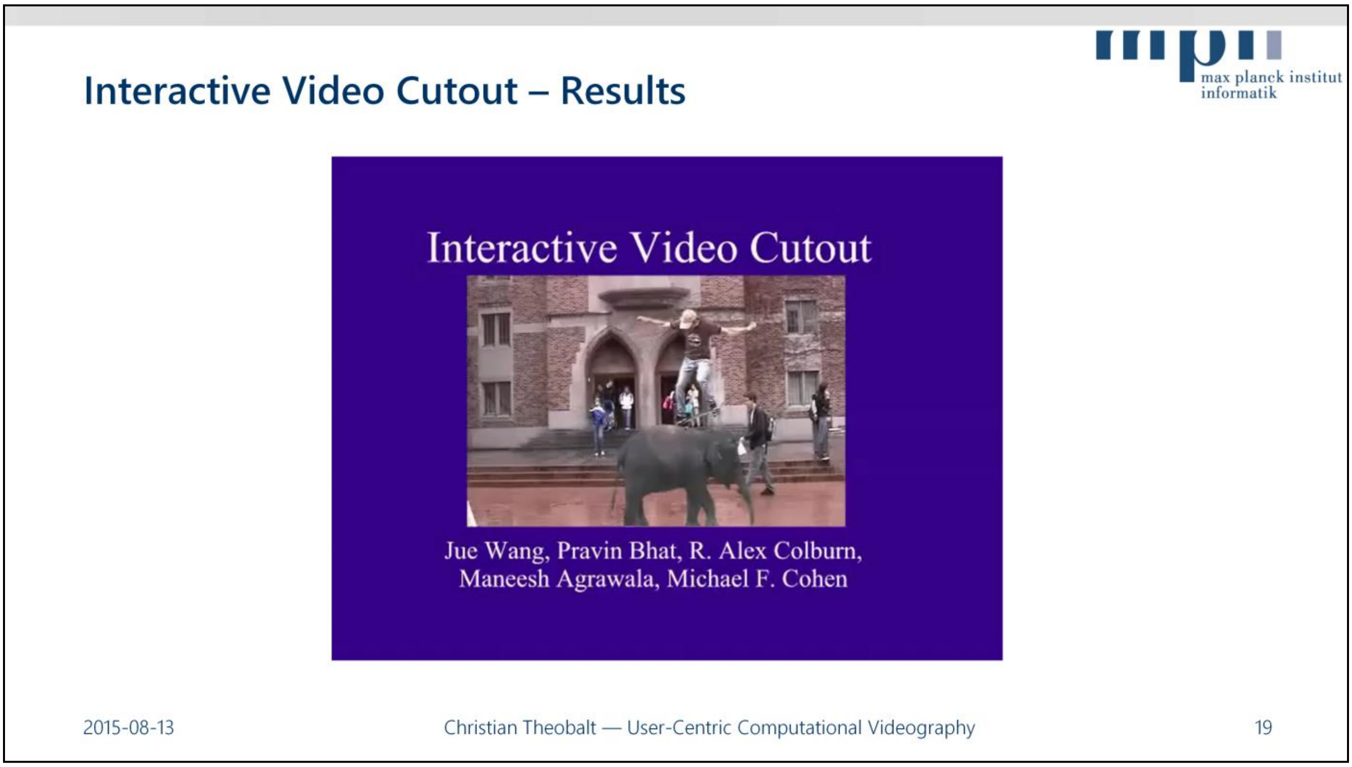
Matting result

[Wang et al., 2005]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 18

The discrete segmentation solution from the previous step is often noisy at the boundaries. Therefore, in a last step, a space-time coherent matting algorithm is applied. To this end, a contour mesh is first fitted to the foreground region in each frame separately, and then the contour curves are linked temporally (using descriptors of contour points based on shape and color information). This yields a space-time coherent contour mesh.

Applying image matting approaches separately to every frame of video leads to temporally unstable results. Therefore, a space-time matting approach is designed. To this end, a trimap area, in which the alpha value profile between foreground and background needs to be solved, is defined in a 10-pixel band that extends to either side around the contour curve. In that band, essentially an extended version of the matting approach from the GrabCut approach of [Rother et al. 2004] is applied, in which an additional temporal term is added such that one solves for the matting simultaneously around all contours in the video.

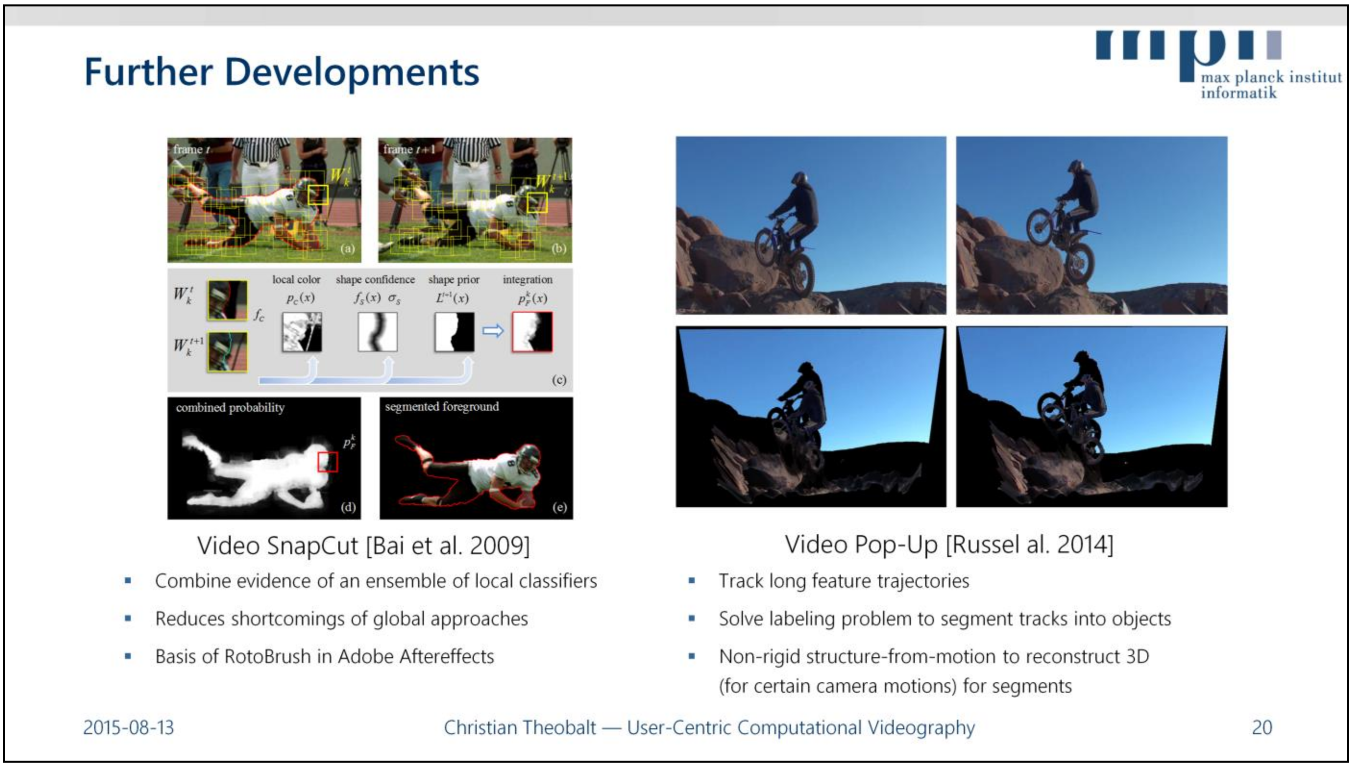


Interactive Video Cutout – Results

Interactive Video Cutout

Jue Wang, Pravin Bhat, R. Alex Colburn, Maneesh Agrawala, Michael F. Cohen

Here are a few results of the approach in action. It enables interactive segmentation of dynamic regions in a video.



Further Developments

Video SnapCut [Bai et al. 2009]

- Combine evidence of an ensemble of local classifiers
- Reduces shortcomings of global approaches
- Basis of RotoBrush in Adobe Aftereffects

Video Pop-Up [Russel al. 2014]

- Track long feature trajectories
- Solve labeling problem to segment tracks into objects
- Non-rigid structure-from-motion to reconstruct 3D (for certain camera motions) for segments

2015-08-13 Christian Theobalt — User-Centric Computational Videography 20

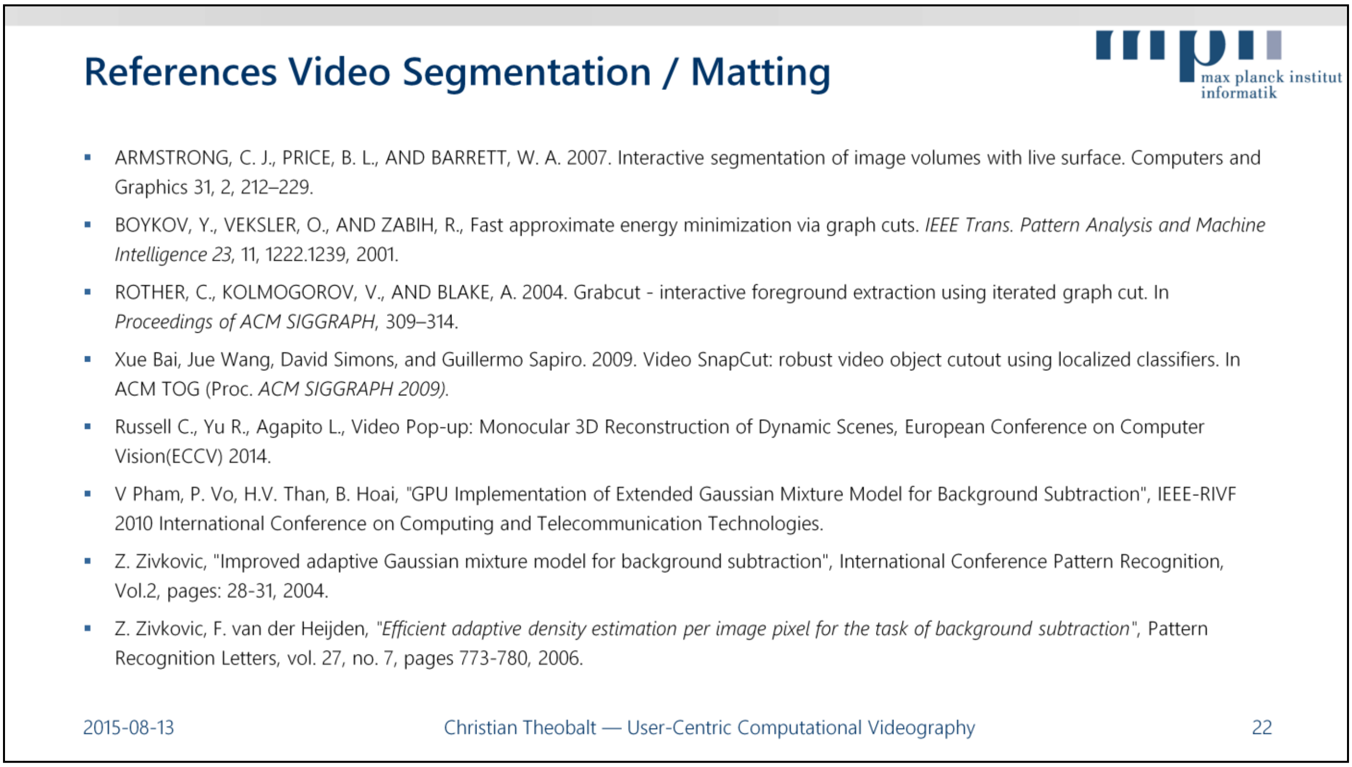
Recently, several improved to video segmentation and matting approaches were proposed. This slides lists two examples.

The Video SnapCut approach [Bai et al. 2009] bypasses certain limitations implied by global approaches, such as Video Cutout. Global approaches are limited by their use of global background or foreground statistics, which can often not properly model local appearance effects. Also global approaches exhibit scalability problems and are not directly applicable for interactive segmentation of long high resolution video clips.Video SnapCut is based on an ensemble of local classifiers that work together to segment foreground and background in space-time coherent way. Each classifier combines evidence from multiple local features, such as color information, edge information, and a shape prior that is learned on-line.

Another recent strand of work combines ideas from non-rigid structure from motion with video segmentation. Russel et al. [2014] extract long feature tracks from videos and solve a labeling problem to segment them into coherent groups that are likely to be coherently moving objects. For each of the objects, a non-rigid-structure-from-motion approach can be applied that estimates the 3D geometry of the deforming segment (along with camera geometry), at least under certain types of relative object –to-camera motion. The end result is a segmentation of moving objects, as well as a 2.5D reconstruction of their shape.

References Video Segmentation / Matting

- ARMSTRONG, C. J., PRICE, B. L., AND BARRETT, W. A. 2007. Interactive segmentation of image volumes with live surface. Computers and Graphics 31, 2, 212–229.
- LI, Y., SUN, J., AND SHUM, H. 2005. Video object cut and paste. In Proc. ACM SIGGRAPH, 595–600.
- WANG, J., AND COHEN, M. 2007. Image and video matting: A survey. Foundations and Trends in Computer Graphics and Vision 3, 2, 97–175.
- WANG, J., BHAT, P., COLBURN, A., AGRAWALA, M., AND COHEN, M. 2005. Interactive video cutout. In Proc. of ACM SIGGRAPH.
- E. Varnousfaderani, M. Gelautz, B. Price, J.H. Cho, Image and Video Matting, Tutorial at ICCV 2013.
- BLAKE, A., AND ISARD, M. 1998. Active Contours. Springer-Verlag.
- CHUANG, Y.-Y., AGARWALA, A., CURLESS, B., SALESIN, D., AND SZELISKI, R. 2002. Video matting of Complex Scenes. In Proc. of ACM SIGGRAPH, 243–248.
- YU, T., ZHANG, C., COHEN, M., RUI, Y., AND WU, Y. 2007. Monocular video foreground/background segmentation by tracking spatial-color Gaussian mixture models. In Proc. of WMVC.
- AGARWALA, A., HERTZMANN, A., SALESIN, D. H., AND SEITZ,S. M. 2004. Keyframe-based tracking for rotoscoping and animation. In Proc. of ACM SIGGRAPH, 584–591.
- Y. Liu, C. Stoll, J. Gall, H.-P. Seidel, C. Theobalt, Markerless Motion Capture of Interacting Characters Using Multi-view Image Segmentation, CVPR 2012.



References Video Segmentation / Matting

- ARMSTRONG, C. J., PRICE, B. L., AND BARRETT, W. A. 2007. Interactive segmentation of image volumes with live surface. Computers and Graphics 31, 2, 212–229.
- BOYKOV, Y., VEKSLER, O., AND ZABIH, R., Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Analysis and Machine Intelligence 23*, 11, 1222.1239, 2001.
- ROTHER, C., KOLMOGOROV, V., AND BLAKE, A. 2004. Grabcut - interactive foreground extraction using iterated graph cut. In *Proceedings of ACM SIGGRAPH*, 309–314.
- Xue Bai, Jue Wang, David Simons, and Guillermo Sapiro. 2009. Video SnapCut: robust video object cutout using localized classifiers. In ACM TOG (Proc. *ACM SIGGRAPH 2009).*
- Russell C., Yu R., Agapito L., Video Pop-up: Monocular 3D Reconstruction of Dynamic Scenes, European Conference on Computer Vision(ECCV) 2014.
- V Pham, P. Vo, H.V. Than, B. Hoai, "GPU Implementation of Extended Gaussian Mixture Model for Background Subtraction", IEEE-RIVF 2010 International Conference on Computing and Telecommunication Technologies.
- Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction", International Conference Pattern Recognition, Vol.2, pages: 28-31, 2004.
- Z. Zivkovic, F. van der Heijden, "*Efficient adaptive density estimation per image pixel for the task of background subtraction*", Pattern Recognition Letters, vol. 27, no. 7, pages 773-780, 2006.

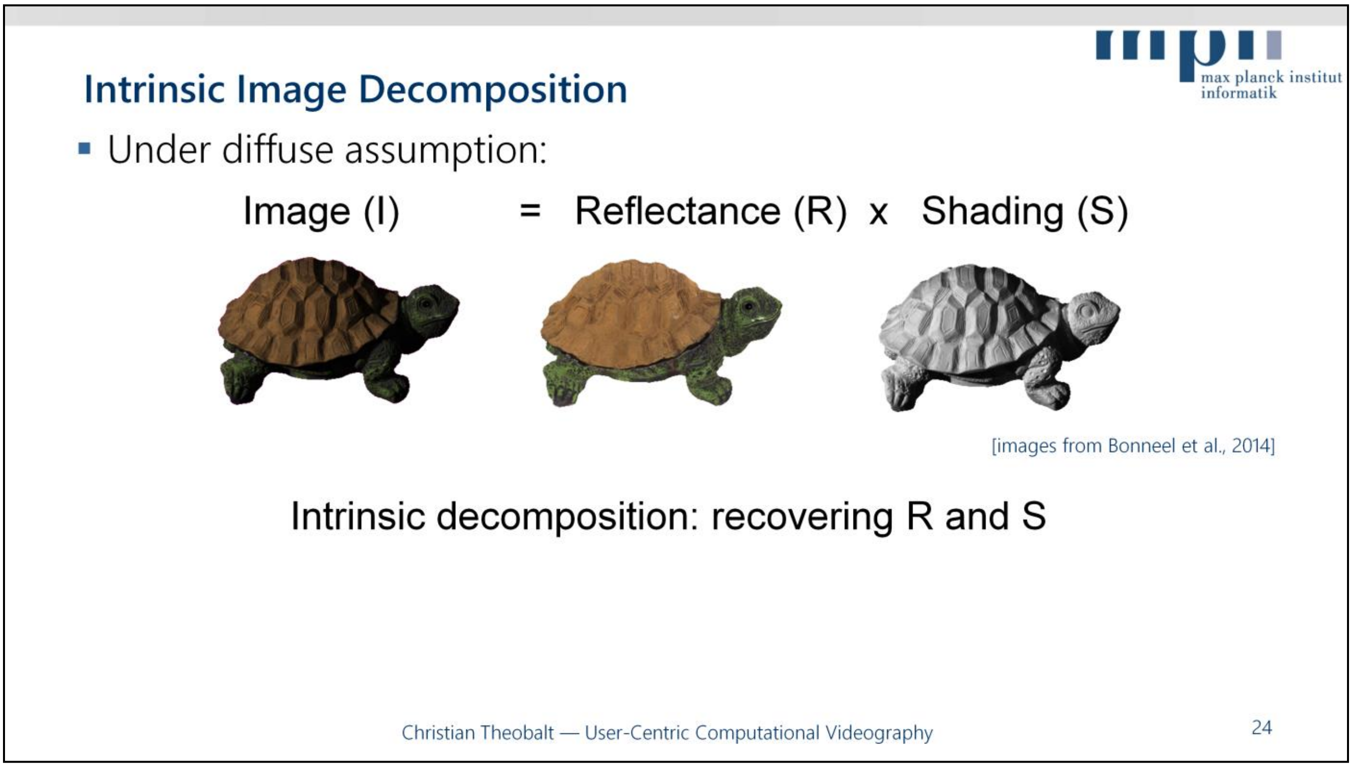
max planck institut informatik

Christian Theobalt

Intrinsic Video

2015-08-13 Christian Theobalt — User-Centric Computational Videography 23

The previous part of the course looked into tools for spatial segmentation of videos into coherent scene entities. The following part looks into a different type of decomposition based on appearance criteria, namely the construction of so-called intrinsic videos.



Intrinsic Image Decomposition

- Under diffuse assumption:

Image (I) = Reflectance (R) x Shading (S)

[images from Bonneel et al., 2014]

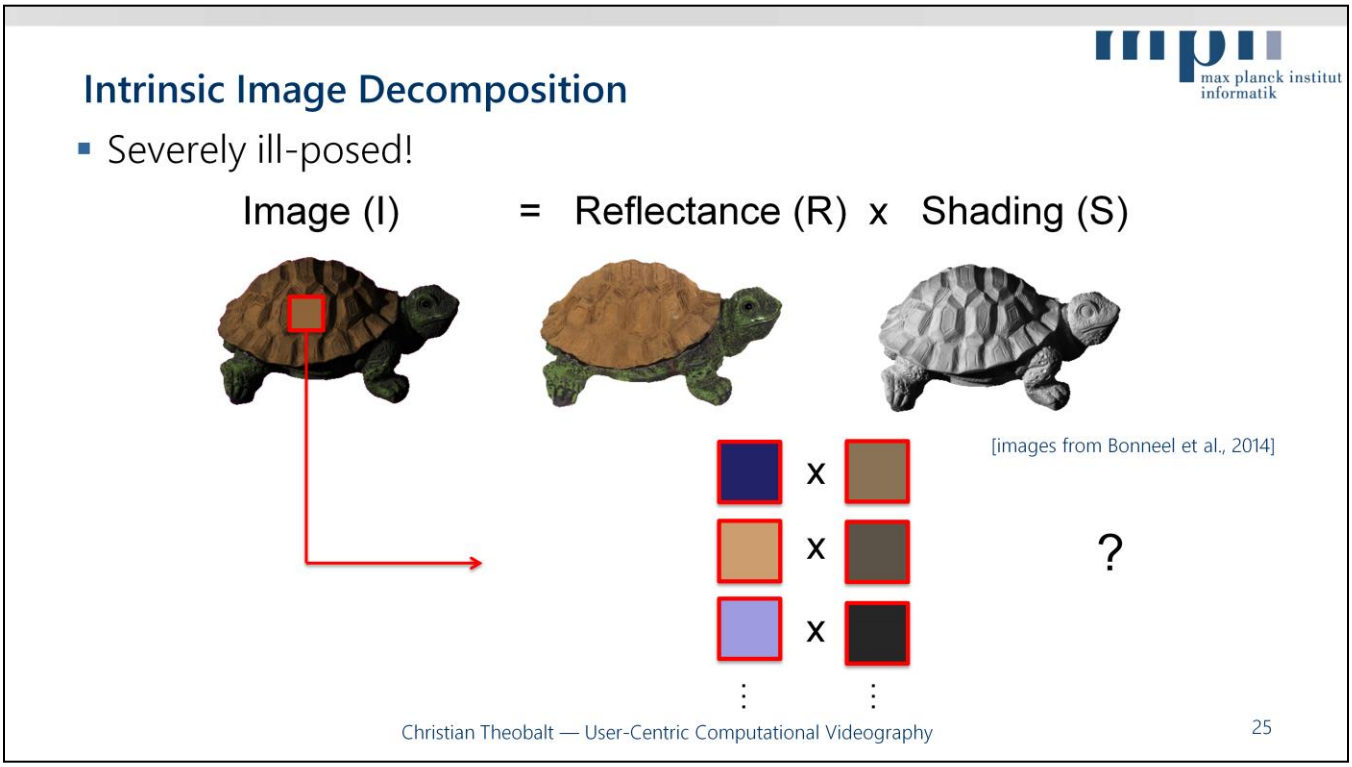
Intrinsic decomposition: recovering R and S

Christian Theobalt — User-Centric Computational Videography 24

As with several other editing operations discussed before, let us first take a look at the single image case to define the problem. Intrinsic image decomposition describes the process of decomposing an image **I** into two specific (scalar) components of appearance, namely a layer describing the reflectance in the scene at every pixel **R**, and a layer describing the illumination or shading in the scene at every pixel **S**. Intrinsic decomposition is a problem that has been studied in computer vision and graphics for many years, and we will, in the following, briefly point to important categories of intrinsic image decomposition approaches. They differ in the way how they methodically approach the fundamental ill-posedness of the problem.

Intrinsic video decomposition, reviewed thereafter, is a computationally even more challenging task, and, as with previous topics covered in this course, is particularly challenging due to the requirement of a temporally coherent decomposition of a sequence of frames. The availability of a sequence of fames may also be advantageous for solving the problem, though, as we will see later.

Side note: The term intrinsic decomposition was coined in the work by Barrow and Tenenbaum [1978]. In their original paper, they define not only reflectance and shading images as the result of intrinsic decomposition, but also additional images related to scene structure, such as normal images, depth images, occluding contour images.



Intrinsic Image Decomposition

- Severely ill-posed!

Image (I) = Reflectance (R) x Shading (S)

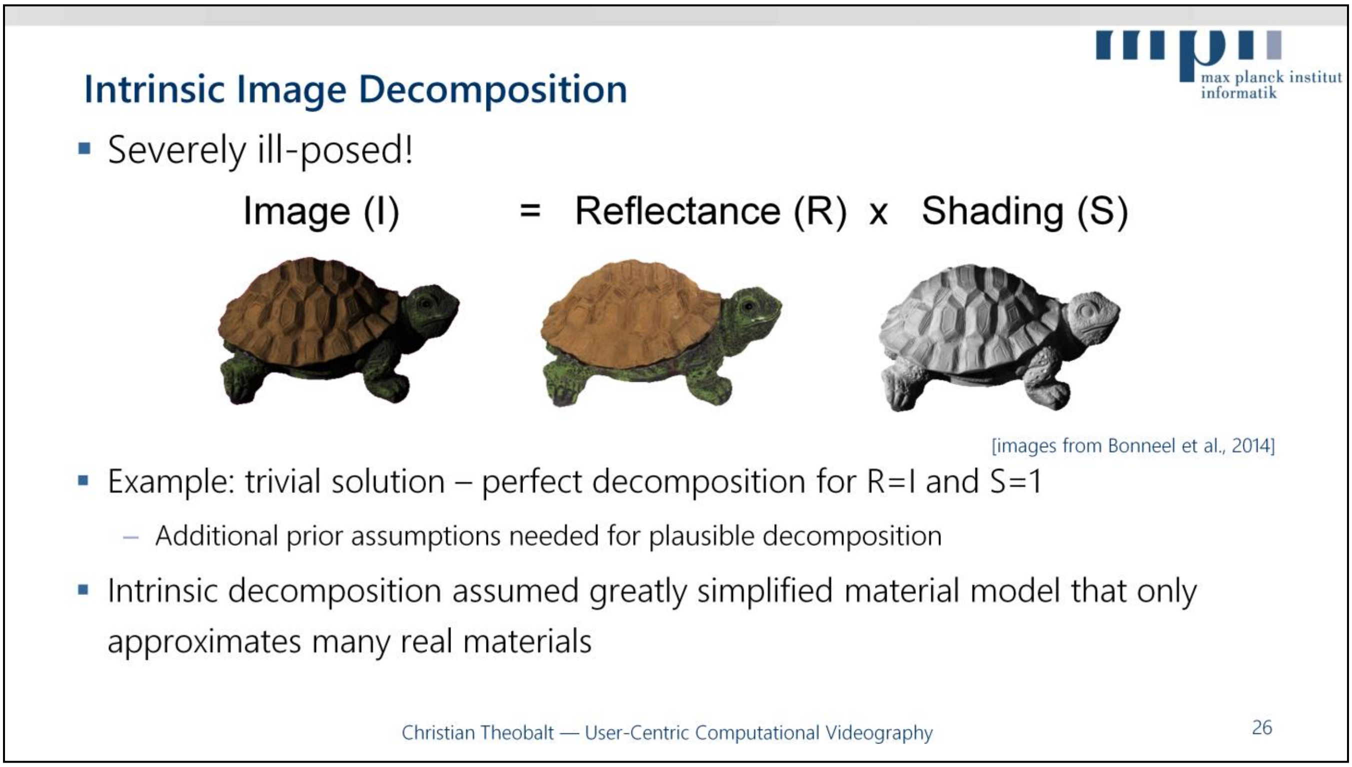
[images from Bonneel et al., 2014]

Christian Theobalt — User-Centric Computational Videography

25

As mentioned on the previous slide, intrinsic decomposition of a single image is an ill-posed problem as there is only one observation per pixel but two unknowns. A possibly infinite number of per-pixel decompositions into shading and reflectance may explain the same image **I**. For instance, even a trivial splitting of an image into a reflectance image **R=I**, and a shading image with all pixel values set to unity **S=1**, are a valid intrinsic decomposition, albeit not a very likely one.

Practical decomposition approaches thus make additional assumptions to differentiate plausible from not plausible decompositions, for instance by enforcing certain priors on shading and reflectance (see discussion of selected related methods).



Intrinsic Image Decomposition

- Severely ill-posed!

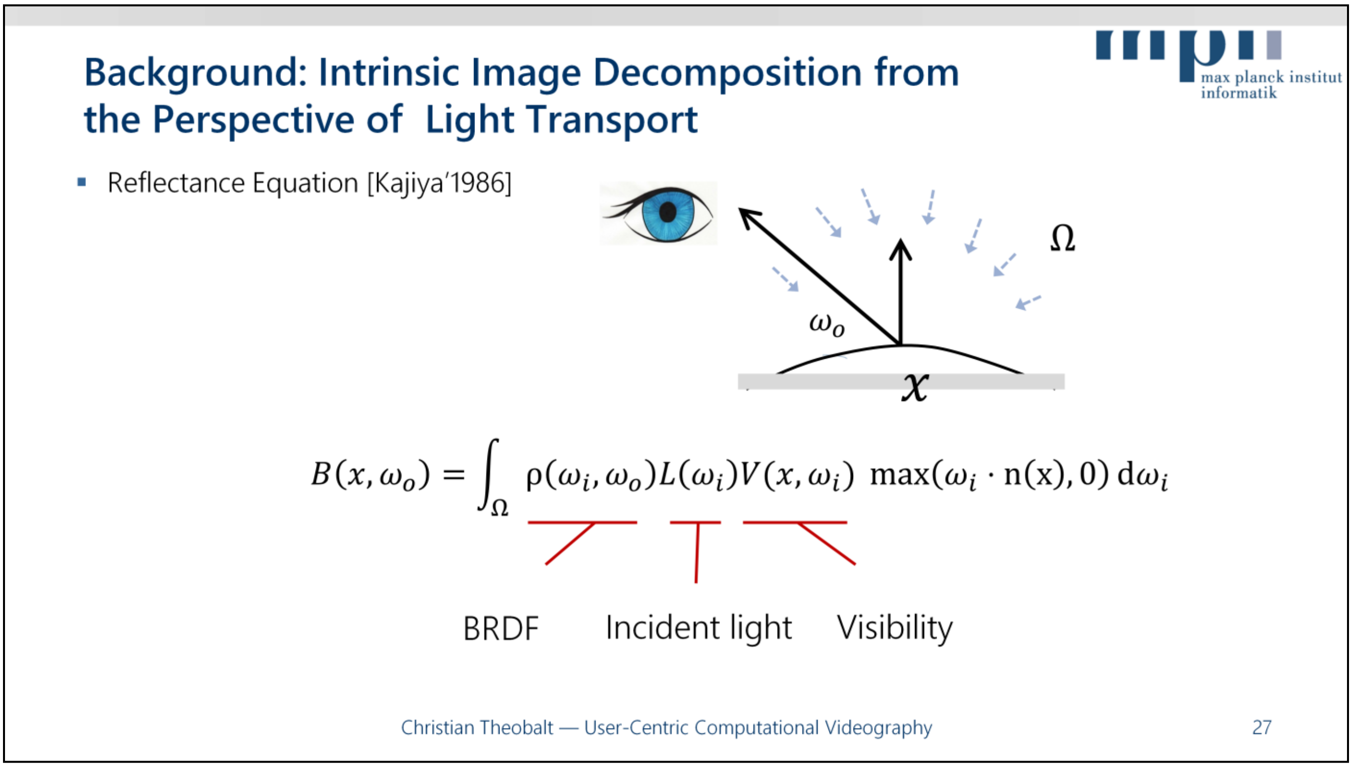
Image (I) = Reflectance (R) x Shading (S)

[images from Bonneel et al., 2014]

- Example: trivial solution – perfect decomposition for R=I and S=1
  - Additional prior assumptions needed for plausible decomposition
- Intrinsic decomposition assumed greatly simplified material model that only approximates many real materials

Christian Theobalt — User-Centric Computational Videography 26

Intrinsic decomposition is a very hard problem, despite the fact that the problem is already phrased under starkly simplifying assumptions about the reality of light transport in a scene, in particular the reflectance of the visible surfaces. In particular, it is assumed that all surfaces in a scene are diffuse, which is often a good approximation to reality, but often also a too strong simplification. In the following slide we take a look at this from the perspective of the rendering / shading equation.



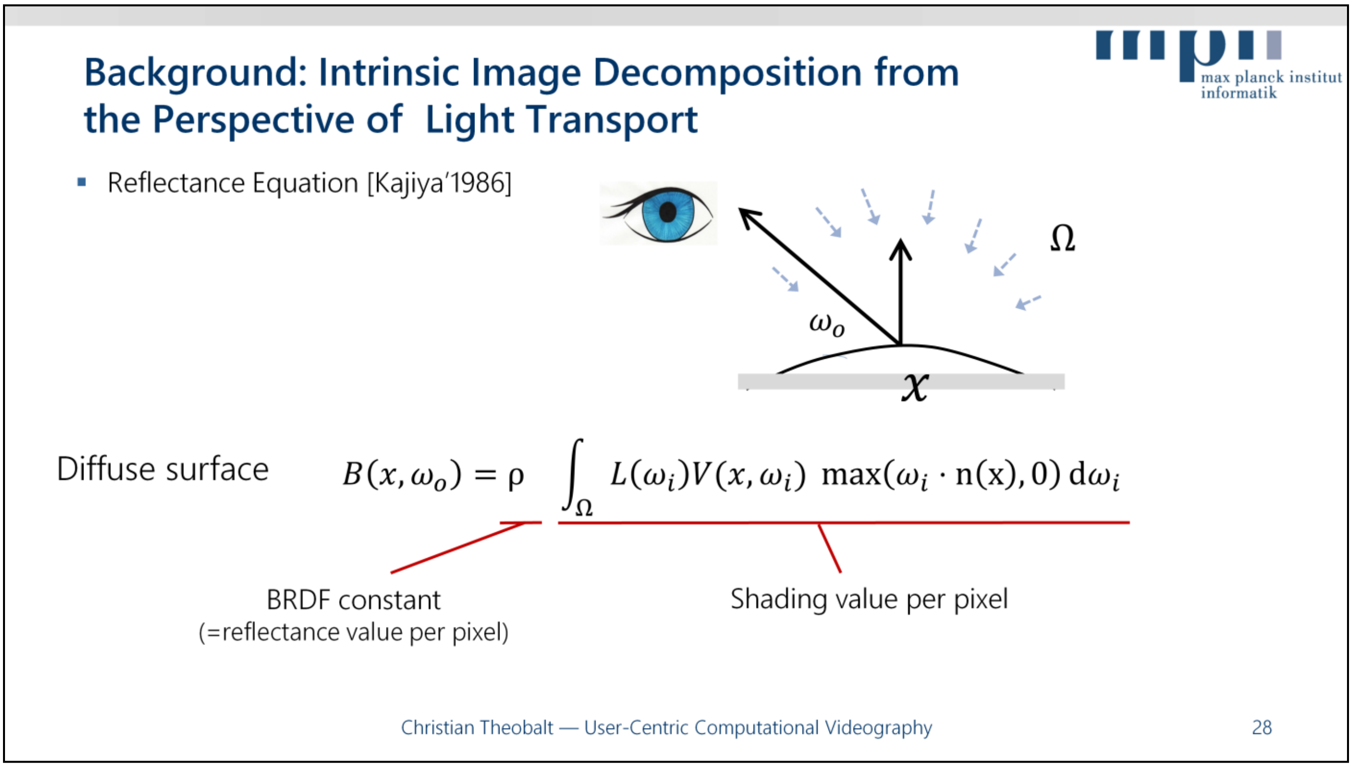
Background: Intrinsic Image Decomposition from the Perspective of Light Transport

- Reflectance Equation [Kajiya'1986]

$$B(x,\omega_o) = \int_{\Omega} \underbrace{\rho(\omega_i,\omega_o)}_{\text{BRDF}}\underbrace{L(\omega_i)}_{\text{Incident light}}\underbrace{V(x,\omega_i)}_{\text{Visibility}}\ \max(\omega_i \cdot \mathrm{n}(\mathrm{x}), 0)\, \mathrm{d}\omega_i$$

Let's briefly take a look at what intrinsic image decomposition means from the perspective of the light transport at surfaces in a scene in 3D, which is described by the reflectance / rendering equation [Kajiya'86]. Please note that the above formulation is also already a simplification of light transport at real surfaces, sub-surface light transport effects are ignored.

This equation describes the outgoing light (radiance) from a point **x** with normal **n** towards the viewer in direction $\boldsymbol{\omega_o}$ if the point is illuminated with incident light (irradiance) **L** from all directions **Ω**. The equation depends on the BRDF (bidirectional reflectance distribution function) of the point describing its reflectance (in particular, the ratio between outgoing radiance to incoming irradiance at the point), as well as the visibility function **V** towards the light source, as well as the cosine of lighting term.



Background: Intrinsic Image Decomposition from the Perspective of Light Transport

- Reflectance Equation [Kajiya'1986]

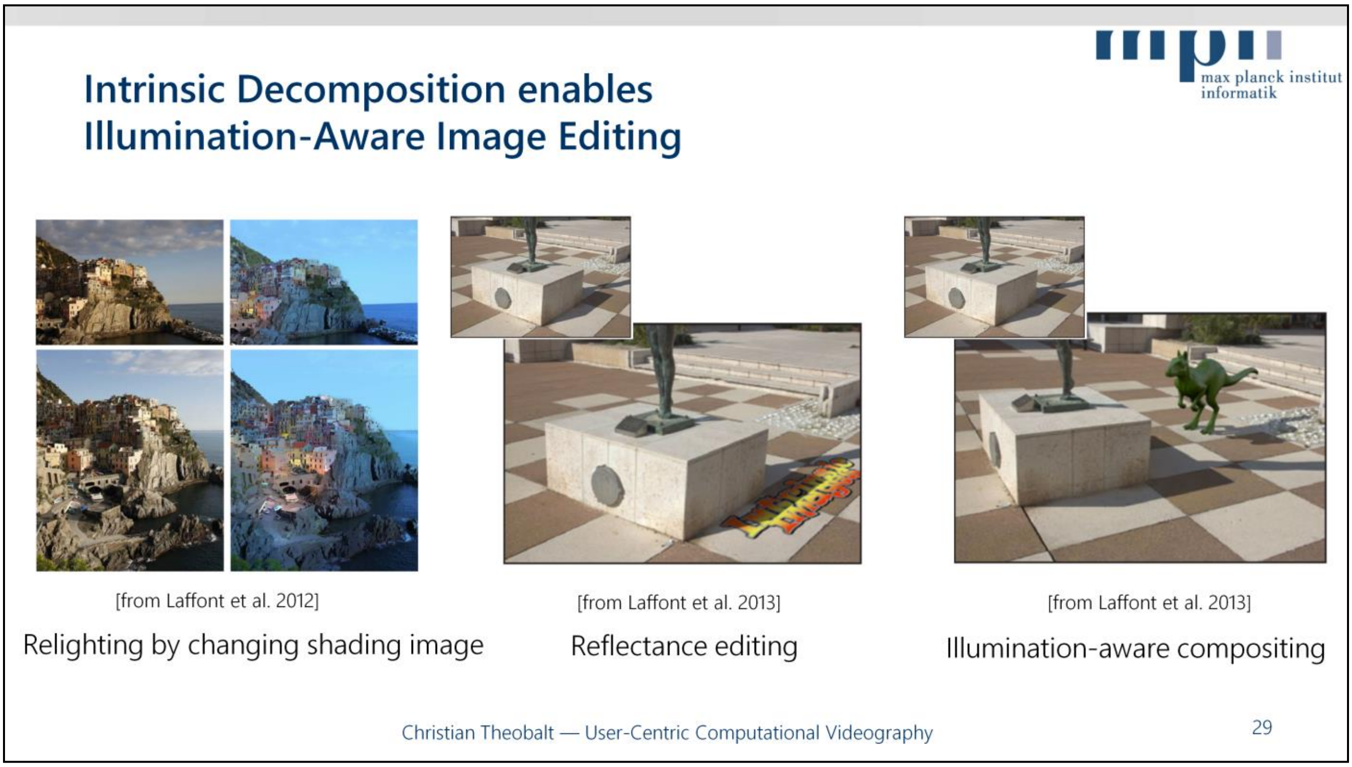
ω_o
Ω
x

Diffuse surface

$$B(x,\omega_o) = \underbrace{\rho}_{\substack{\text{BRDF constant}\\ \text{(=reflectance value per pixel)}}} \underbrace{\int_{\Omega} L(\omega_i)V(x,\omega_i)\ \max(\omega_i \cdot \mathrm{n(x)}, 0)\, \mathrm{d}\omega_i}_{\text{Shading value per pixel}}$$

Intrinsic image decomposition assumes that all surfaces in a scene have a Lambertian (diffuse) reflectance. Under the this assumption, the BRDF is a constant and can be factored out of the integral. An intrinsic image thus decomposes the scene per pixel into the constant BRDF value, i.e. the diffuse albedo, and the shading value which is the complete integral on the right hand side evaluated at that pixel.

Obviously, the reflectance of most real-world materials is not purely Lambertian, and also exhibits more advanced and higher frequency effects, such as specularities, sub-surface lighting effects, etc. This may lead to artefacts in the decomposition. However, most materials indeed do have a strong diffuse component in their reflectance, and intrinsic decomposition thus provides a good approximation to the real appearance of many materials.



Intrinsic Decomposition enables Illumination-Aware Image Editing

[from Laffont et al. 2012]

Relighting by changing shading image

[from Laffont et al. 2013]

Reflectance editing

[from Laffont et al. 2013]

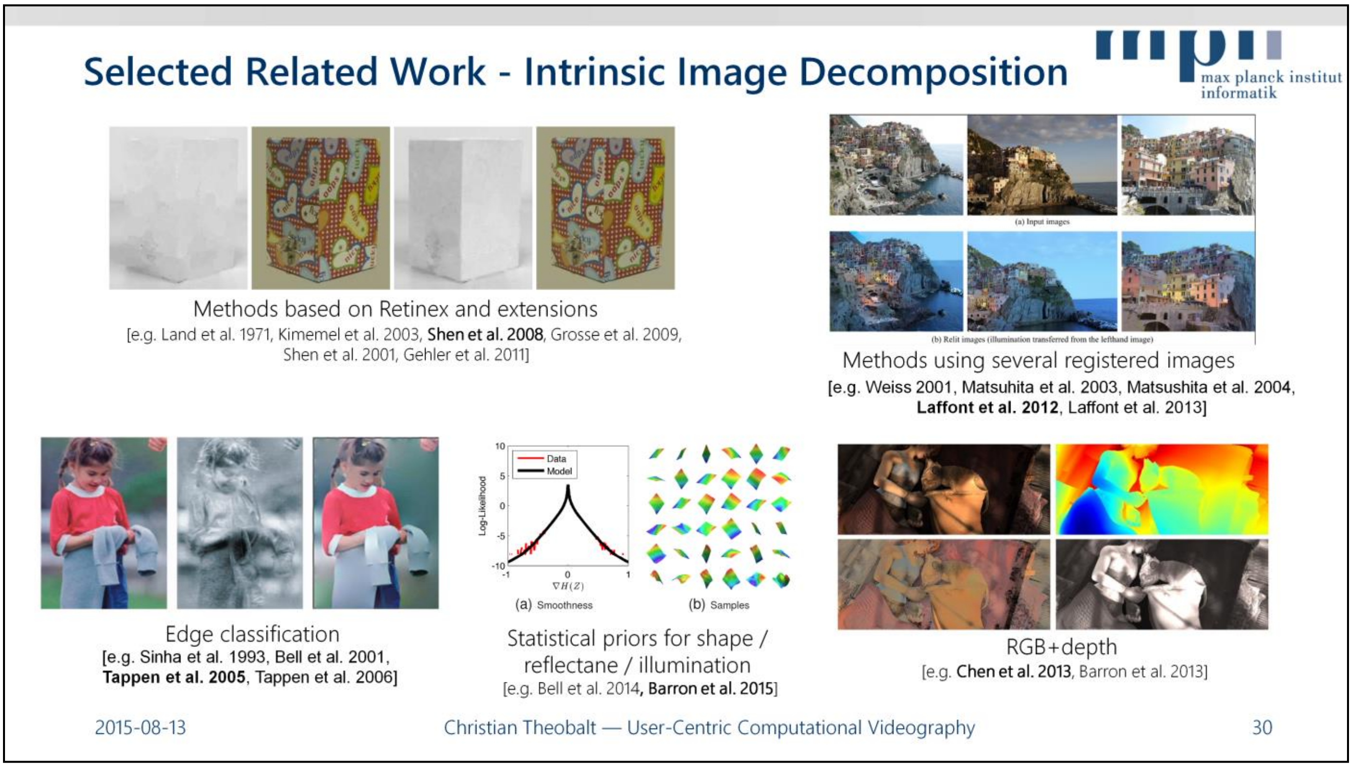
Illumination-aware compositing

Christian Theobalt — User-Centric Computational Videography

29

What is the benefit of an intrinsic frame decomposition from the perspective of image or video editing? Essentially, given the decomposition of an image into per-pixel reflectance and shading, one can perform illumination-aware image modifications by modifying the individual layers of the decomposed image and then recombining the layers to get the modified composite image. The above images show three such editing examples from related work.

For instance, one can modify the scene illumination by editing the shading layer per pixel and thus simulate how a scene would look like under a different lighting condition. One can also edit the reflectance of a scene, and thus create properly lit newly textured surfaces, such as the writing on the floor in the center image above. One can also simulate a new 3D object being composited into the scene and approximate its appearance under the scene lighting, if one respects the proper 3D positioning and camera projection geometry.



Selected Related Work - Intrinsic Image Decomposition

Methods based on Retinex and extensions
[e.g. Land et al. 1971, Kimemel et al. 2003, **Shen et al. 2008**, Grosse et al. 2009, Shen et al. 2001, Gehler et al. 2011]

Methods using several registered images
[e.g. Weiss 2001, Matsuhita et al. 2003, Matsushita et al. 2004, **Laffont et al. 2012**, Laffont et al. 2013]

Edge classification
[e.g. Sinha et al. 1993, Bell et al. 2001, **Tappen et al. 2005**, Tappen et al. 2006]

Statistical priors for shape / reflectane / illumination
[e.g. Bell et al. 2014**, Barron et al. 2015**]

RGB+depth
[e.g. **Chen et al. 2013**, Barron et al. 2013]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 30

Given an image, intrinsic decomposition computes - per pixel - two scalar unknowns, reflectance and shading. Usually, this is formulated as some form of energy minimization problem. As stated before, intrinsic image decomposition is a fundamentally ill-posed problem. Therefore, previous methods resorted to a variety of additional assumptions in order to guide the decomposition towards plausible solutions.

Several approaches make prior assumptions about material or lighting statistics which can be used as priors in intrinsic image decomposition. The Retinex theory by Land and McCann [Land et al., 1971] inspired many intrinsic decomposition works [Kimmel et al. 2003, Grosse et al., 2009, Shen et al. 2008]. Under Retinex, small gradients are assumed to correspond to illumination and large gradients are assumed to correspond to reflectance. The theory is only an approximation to reality and developed for a Mondrian (piecewise constant) world. It has been combined with several additional cues to achieve better decompositions, e.g. non-local texture cues [Shen et al. 2008], or global sparsity priors (Shen et al. 2001, Gehler et al. 2011].

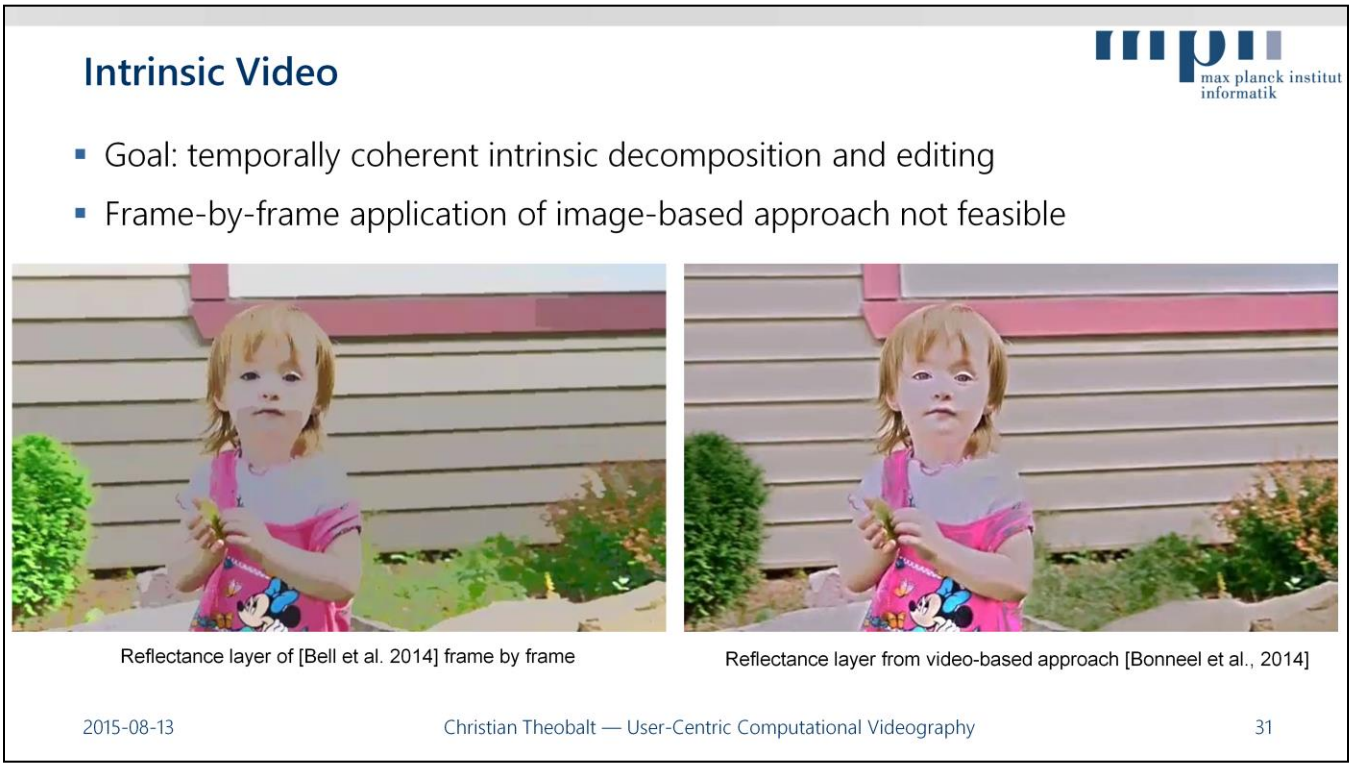
Other approaches aim for more reliable decompositions by try to classifying edges into illumination edges and reflectance edges [e.g. Sinha et al. 1993, Bell et al. 2001, Tappen et al. 2005, Tappen et al. 2006]

Intrinsic decomposition is also simplified by considering information not only from one image but from multiple registered images, for instance images taken under different illumination conditions [Weiss 2001, Matsushita et al. 2003, Matsushita et al. 2004, Laffont et al. 2013], or images from a community image collection from which a sparse 3D model is reconstructed via structure-from motion. In this case, additional cross-image constraints can be formulated [Laffont et al. 2012].

Recent work also investigated the use of statistical priors on shape, reflectance and illumination for improved decomposition, e.g. [Barron et al., 2015]. The lack of extensive real world data sets with proper ground truth annotations of reflectance and shading has also been identified as a bottleneck hindering innovation. Bell et al. [2014] thus provide a large data set of real world images for which intrinsic image annotations were created through crowdsourcing. They also propose a new intrinsic image decomposition algorithm based on a conditional random field in which also longer range relationships, such as reflectance similarities between different parts of the image, can be encoded.

New consumer grade RGBD sensors offer additional possibilities for intrinsic image decomposition. Some approaches therefore employ per-pixel depth information in addition to per pixel color information for intrinsic image decomposition, e.g. [Chen et al. 2013, Barron et al. 2013].

Most decomposition methods for single images or a few images cannot be simply applied to each frame of video. This would yield temporally incoherent solutions. Therefore, dedicated video-based decomposition approaches were researched.



Intrinsic Video

- Goal: temporally coherent intrinsic decomposition and editing
- Frame-by-frame application of image-based approach not feasible

Reflectance layer of [Bell et al. 2014] frame by frame

Reflectance layer from video-based approach [Bonneel et al., 2014]

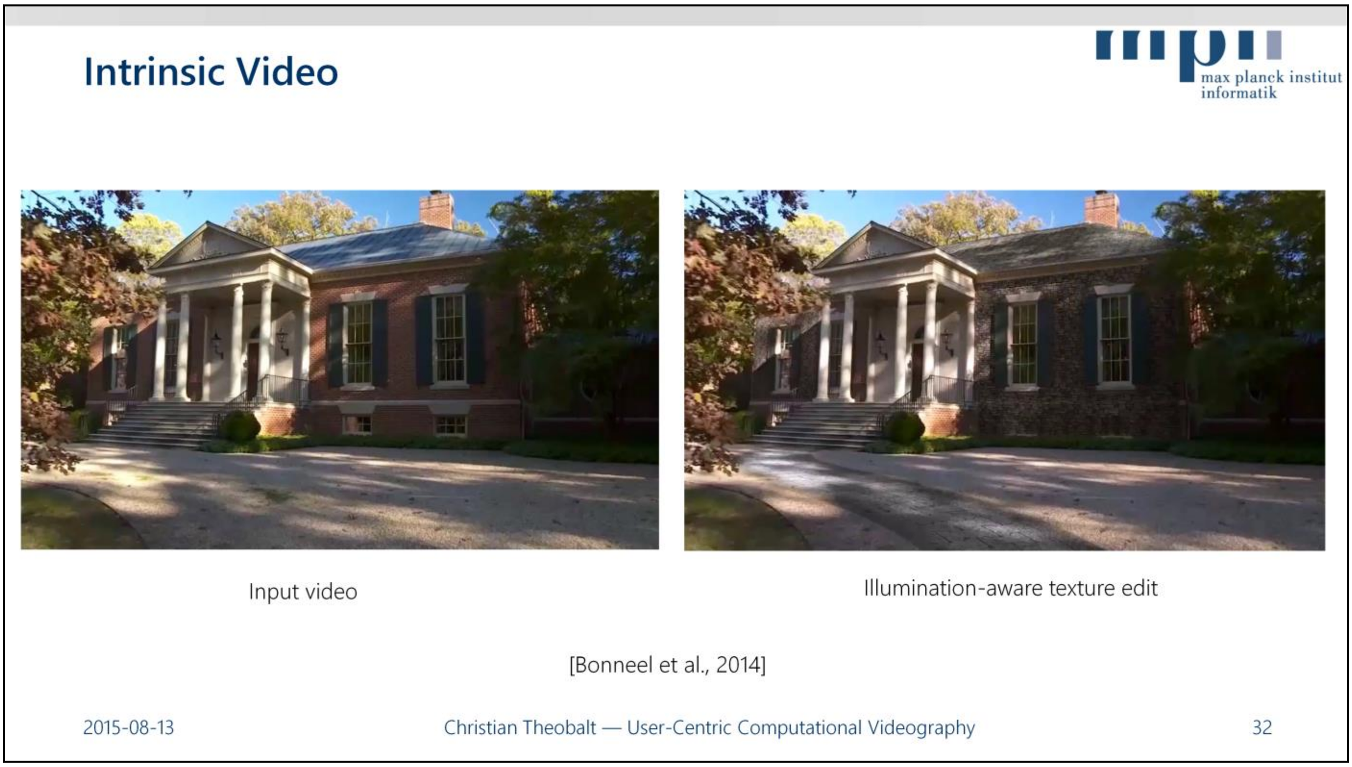
2015-08-13 Christian Theobalt — User-Centric Computational Videography 31

As with many other types of effects discussed in this course, straightforward application on an intrinsic image decomposition approach to each frame of video does usually not lead to satisfactory results. The left video shows an example of applying the approach by Bell et at. [2014] to a video which leads to temporal artefacts.

Temporal coherence in the result needs to be explicitly considered in the algorithm. This is shown in the video on the right shows a result with the specially tailored video-based approach by Bonneel et al. [2014] which leads to a much more stable decomposition.

The field of intrinsic video decomposition is still in its early stage. In the following we will briefly review the concepts behind the approach of Bonneel et al., and also briefly point to related video-based methods.

In addition to producing incoherent results, the use of many intrinsic image decomposition approaches to every frame of a video is computationally prohibitive as results. For even short sequences would not be obtained at acceptable computation times. The approach by Bonneel et al. therefore also addresses the performance question; it enables interactive intrinsic video decomposition with user guidance.



Intrinsic Video

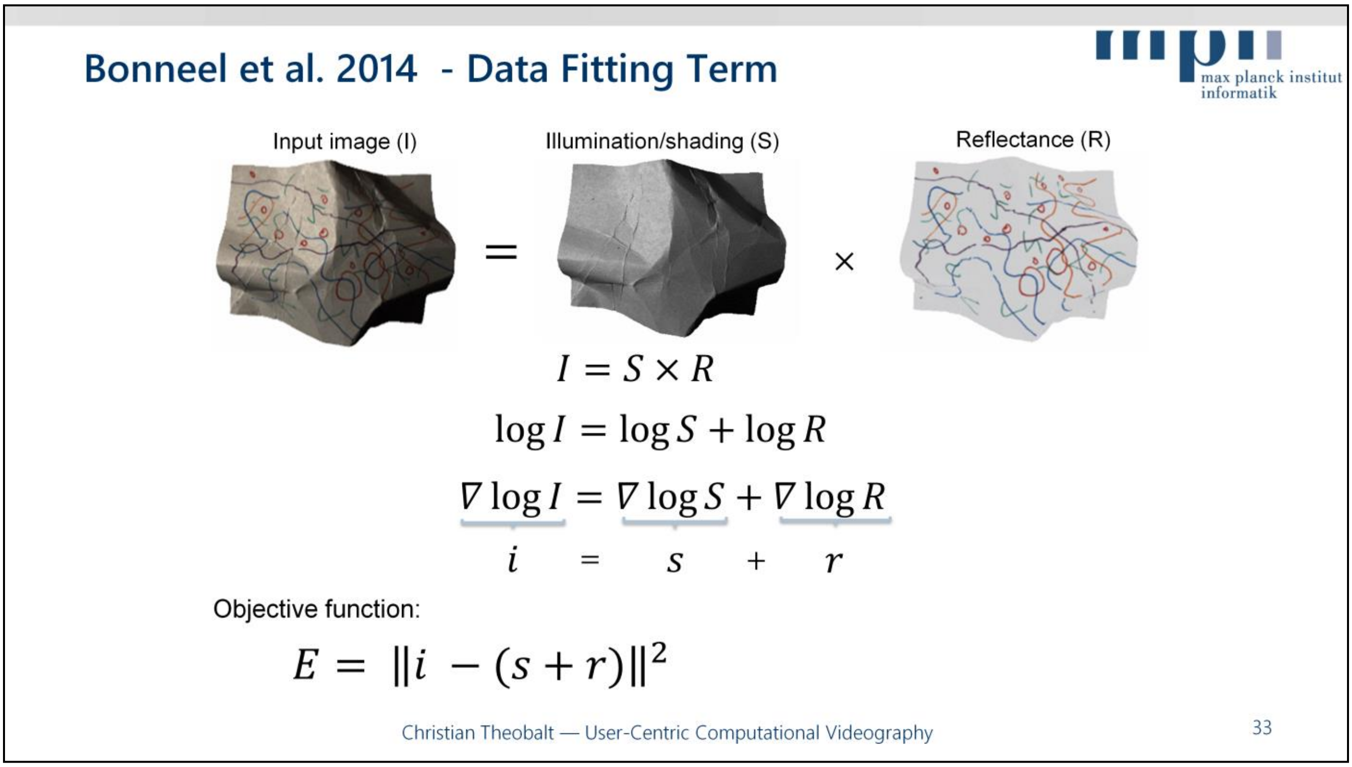
Input video

Illumination-aware texture edit

[Bonneel et al., 2014]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 32

The approach by Bonneel et al. [2014] also enables space-time coherent and illumination-aware modification of the appearance of certain scene elements in a video, as shown for the texture on the wall of this house. To create such a result, correspondences between scene elements need to be tracked accurately over time which may not be feasible for all types of scenes with complex dynamics. For certain types of geometry, such as a planar regions, stable tracking is feasible.



Bonneel et al. 2014 - Data Fitting Term

Input image (I) = Illumination/shading (S) × Reflectance (R)

$$I = S \times R$$

$$\log I = \log S + \log R$$

$$\underbrace{\nabla \log I}_{i} = \underbrace{\nabla \log S}_{s} + \underbrace{\nabla \log R}_{r}$$

Objective function:

$$E = \|i - (s + r)\|^2$$

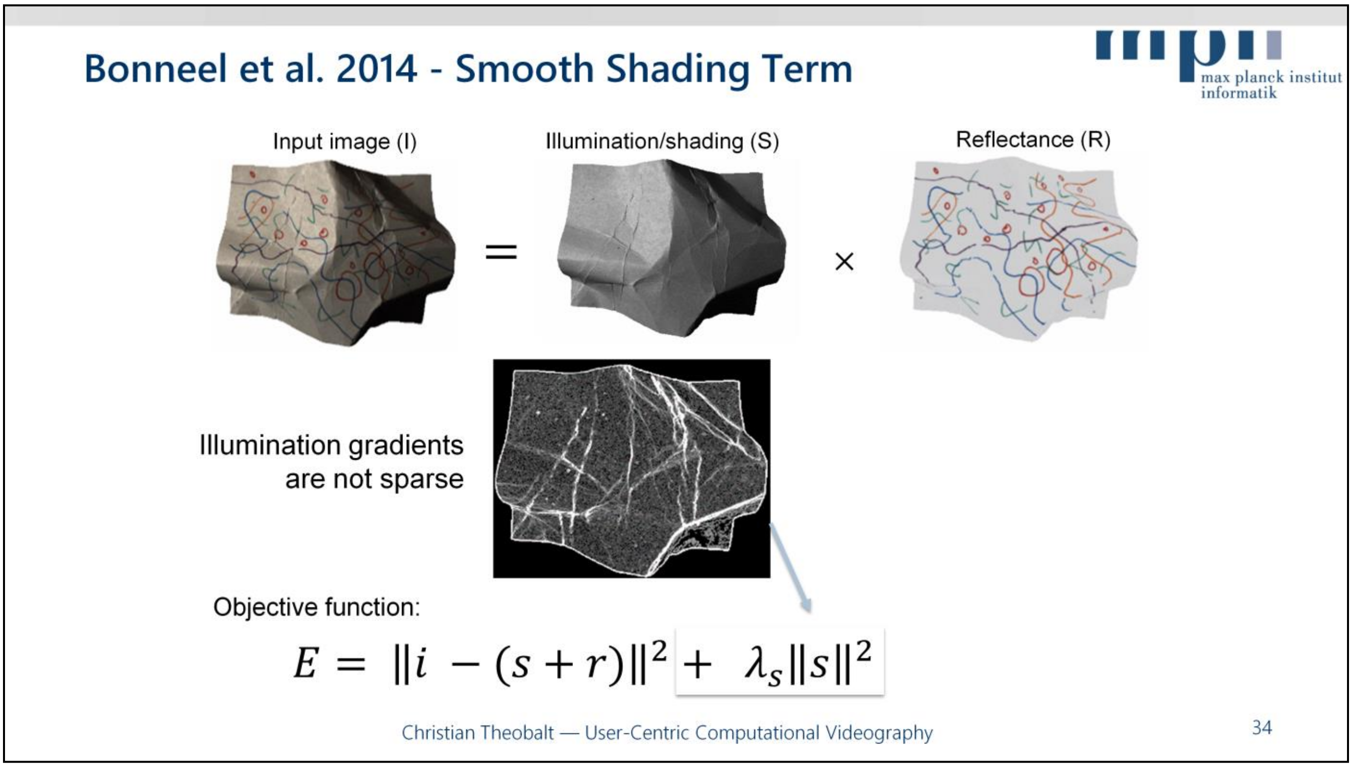
Christian Theobalt — User-Centric Computational Videography

33

Bonneel et al. formulate intrinsic video decomposition as the minimization of an energy function whose components we look at on the following slides. It is assumed that the lighting in the scene is monochromatic. For efficiency, most computations are performed on single channel luminance images. The image luminance I=($\mathbf{I}_r$,$\mathbf{I}_g$,$\mathbf{I}_b$)$^{1/3}$ is defined as the geometric mean of the individual per pixel RGB components. Similarly, the reflectance *R* is defined as the geometric mean of the respective reflectance components. The color reflectance **R** at pixel x can be computed from *R* as **R**(x)=**I**(x)/R(x).

Further on, the problem is defined in the log domain which transforms image formation into a sum. This has the additional advantage that one does not explicitly have to constrain the solutions to be non-negative (as reflectance and shading should be).

The approach is further formulated in the gradient domain. The idea is to separate image gradients into reflectance gradients that are assumed to be spatially sparse, and shading gradients that are supposed to be smooth. In the above formulation, *i*,*s*, and *r* are the log gradients of the image, reflectance and shading,respectively. The first term in the energy is thus a least-squares data term that follows from the image formation model.



Bonneel et al. 2014 - Smooth Shading Term

Input image (I) = Illumination/shading (S) × Reflectance (R)

Illumination gradients are not sparse

Objective function:

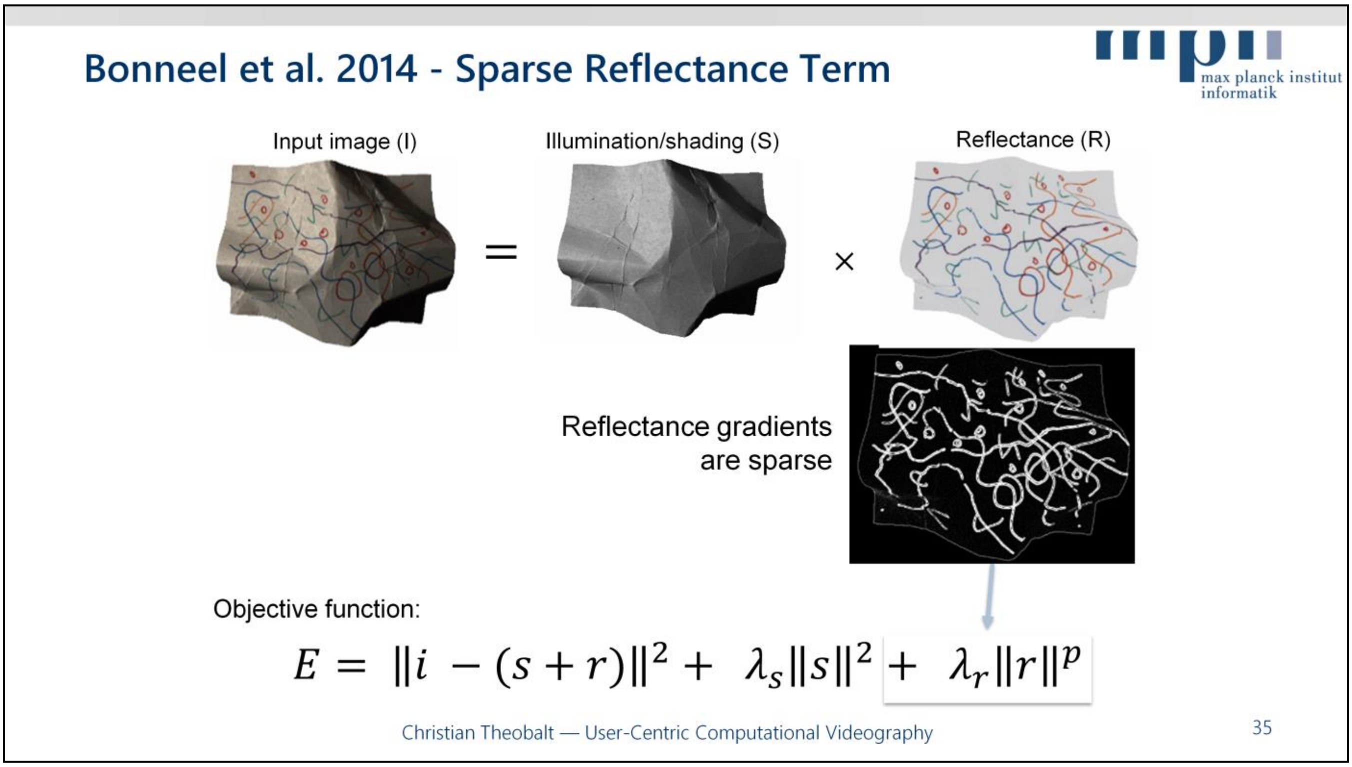
$$E = ||i - (s + r)||^2 + \lambda_s ||s||^2$$

Christian Theobalt — User-Centric Computational Videography

34

This least squares data term alone is ambiguous since only *i* is known, and both *s* and *r* are unknown. Therefore, additional prior terms on *s* and *r* are added to the energy.

First, it is assumed that illumination (shading) exhibits smooth variations, for instance due to illumination on smooth surfaces or soft shadows (see Retinex theory [Land et al. 1971]). This is modeled with an L2-term on the shading gradients *s*.



Bonneel et al. 2014 - Sparse Reflectance Term

Reflectance gradients are sparse

Objective function:

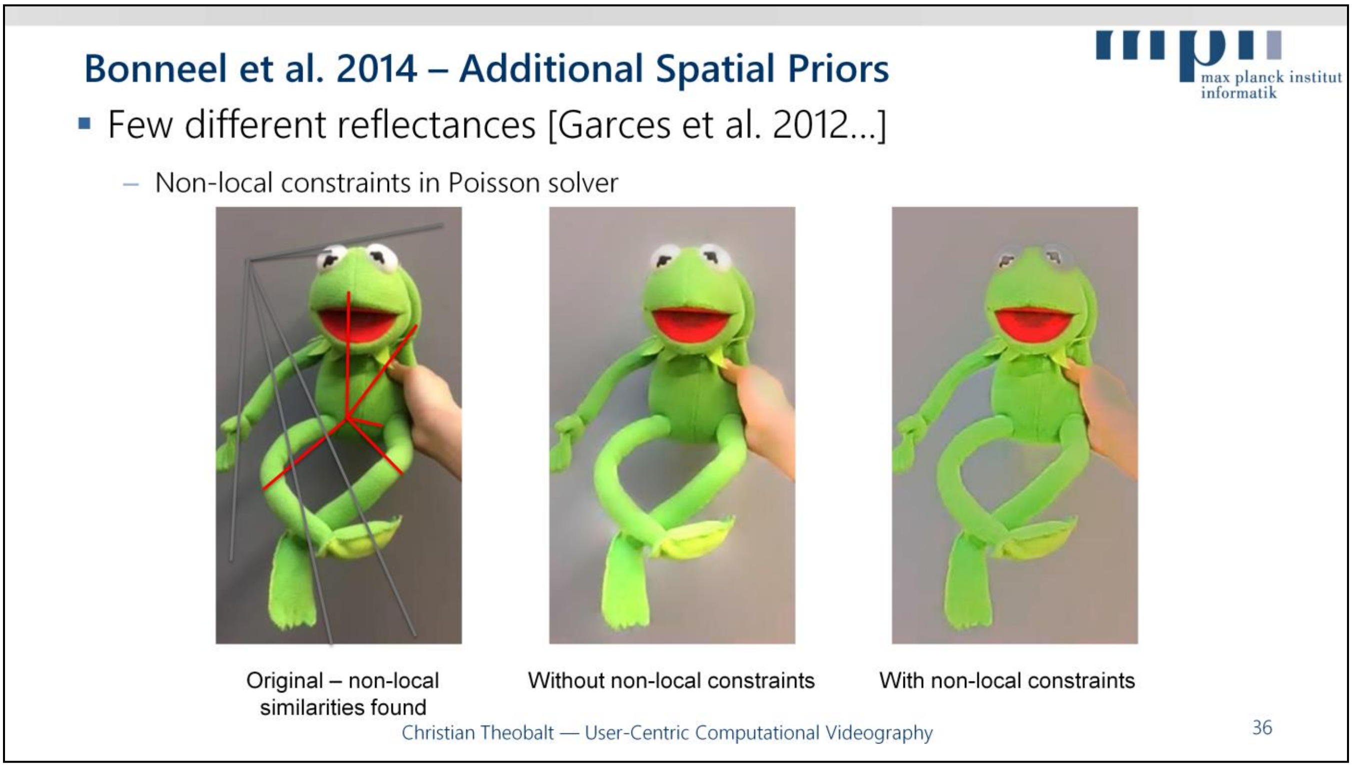
$$E = \|i - (s + r)\|^2 + \lambda_s \|s\|^2 + \lambda_r \|r\|^p$$

Christian Theobalt — User-Centric Computational Videography

35

Second, it is assumed that reflectance values are sparse [Omer et al. 2004, Hsu et al. 2008], i.e. that scenes are mostly made of objects with constant colors separated by hard boundaries. This is modelled using an **Lp** norm on the reflectance gradients with p<2. Low p values enforce very sparse reflectance gradients over the image (In the above energy function, the dependencies on the pixel positions are dropped for better readability, but *s* and *r* are solved for at every pixel).

The energy function developed on the previous slides requires an efficient minimization of a mixed L2-Lp norm energy. Usually, such mixed norm energies are optimized with expensive combinatorial approaches.
In contrast, in this particular case much higher computational efficiency can be achieved by using a look-up table within an iterative reweighthed least squares approach that replace the Lp term with a reweighted L2 term. This yields an iterative update for the reflectance estimate as described above. The details of this proposed minimization algorithm are discussed in the paper by Bonneel et al. [2014].



Bonneel et al. 2014 – Additional Spatial Priors

- Few different reflectances [Garces et al. 2012…]
  - Non-local constraints in Poisson solver

Original – non-local similarities found

Without non-local constraints

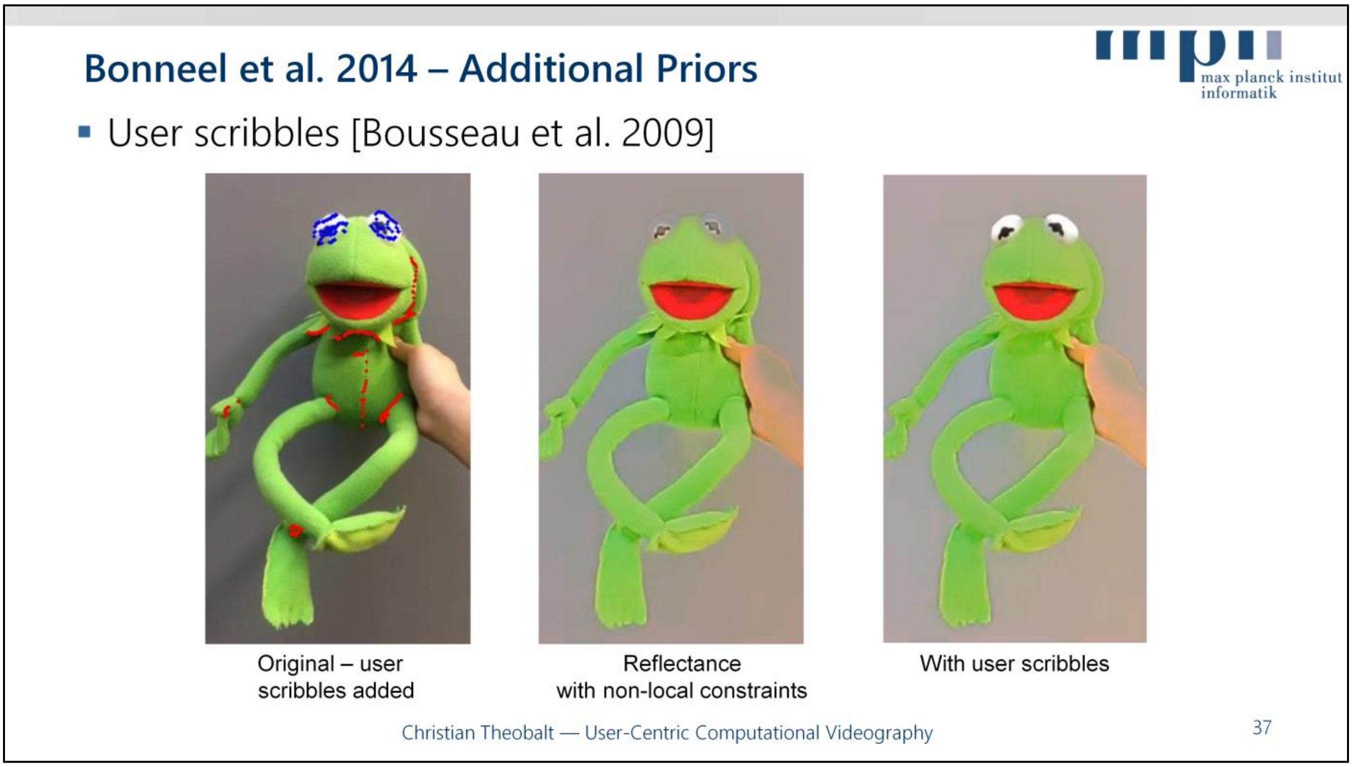
With non-local constraints

Christian Theobalt — User-Centric Computational Videography

36

The solutions obtained with the previous L2-Lp optimization may exhibit low frequency reconstruction errors. In previous work, this problem has been approached by enforcing non-local reflectance constraints, for instance by stating that pixels with similar chrominance should have similar reflectance, e.g. [Shen et al. 2011]. Introducing these non-local constraints leads to dense linear systems and lower computational performance.

To maintain computational efficiency, Bonneel et al. introduce non-local constraints by clustering chrominance into k clusters and enforcing reflectance similarity at each pixel to a sparse set of nearby representative pixels from each cluster that is similar in chrominance.



Bonneel et al. 2014 – Additional Priors

- User scribbles [Bousseau et al. 2009]

Original – user scribbles added

Reflectance with non-local constraints

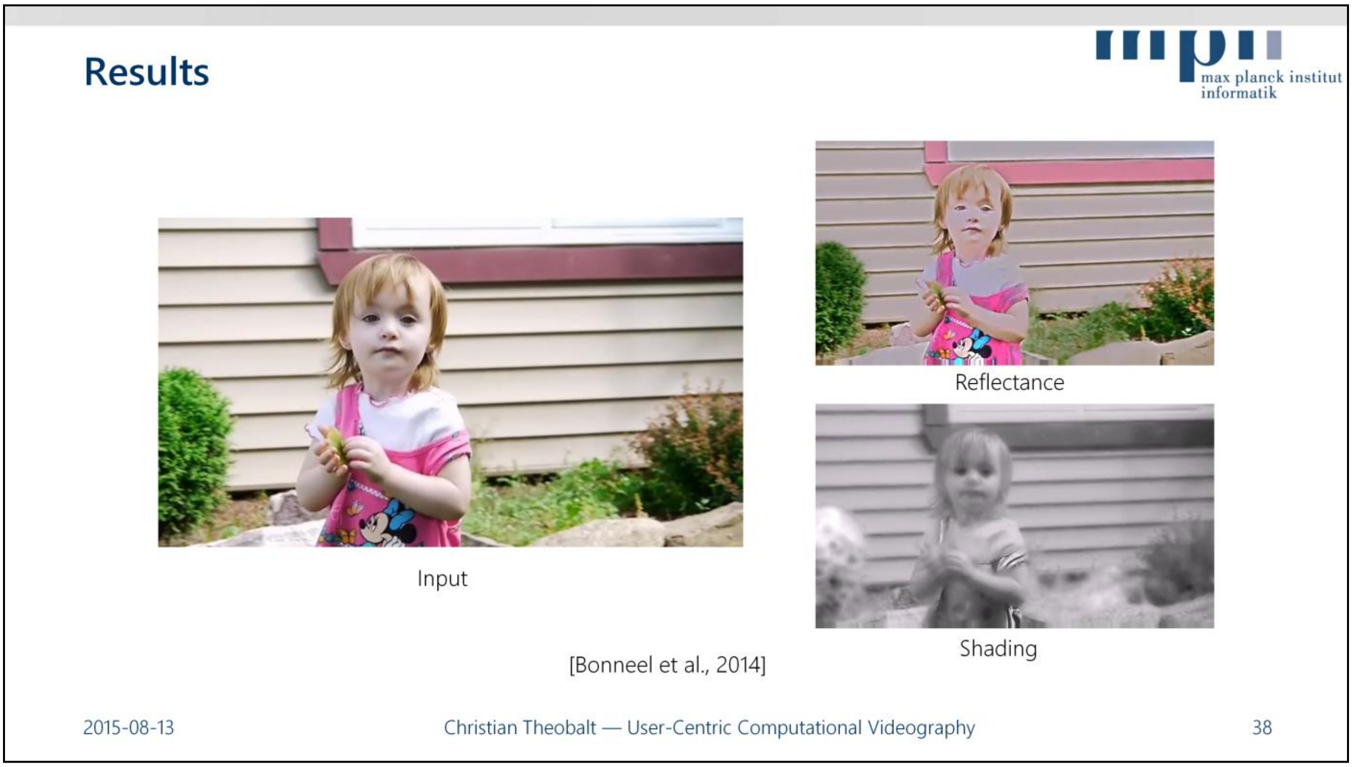
With user scribbles

Christian Theobalt — User-Centric Computational Videography

37

Previous work in intrinsic image decomposition has shown the benefit of constraints from user guidance in resolving decomposition ambiguities [Bousseau et al. 2009].

This video-based approach thus enables the user to provide guidance to the algorithm by means of strokes in order to enable more reliable decomposition. Specifically, two types of strokes can be provided, namely strokes indicating regions of constant reflectance and regions of constant shading. User strokes applied to individual frames are automatically propagated forward in time.



Results

Input

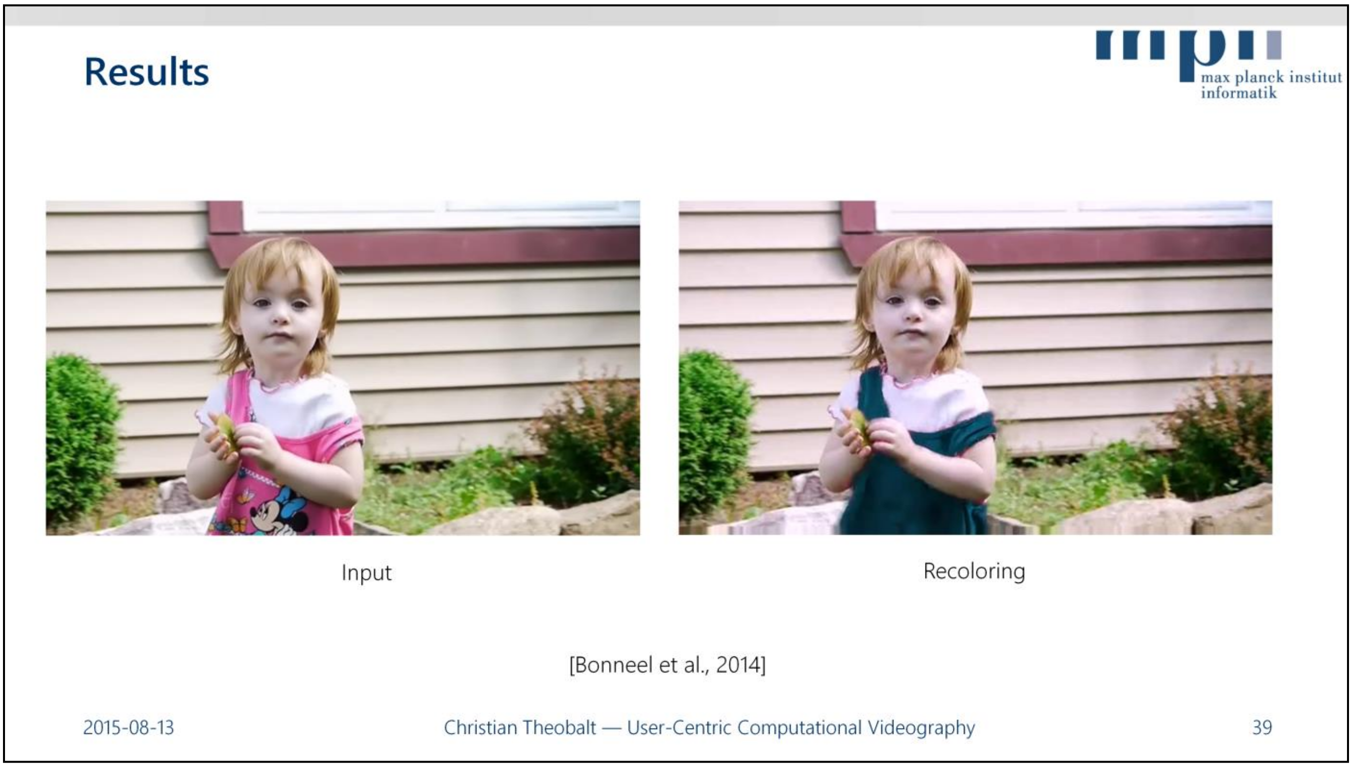
Reflectance

Shading

[Bonneel et al., 2014]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 38

This result video shows a decomposition of the video on the left into reflectance and shading videos.



Results

max planck institut informatik

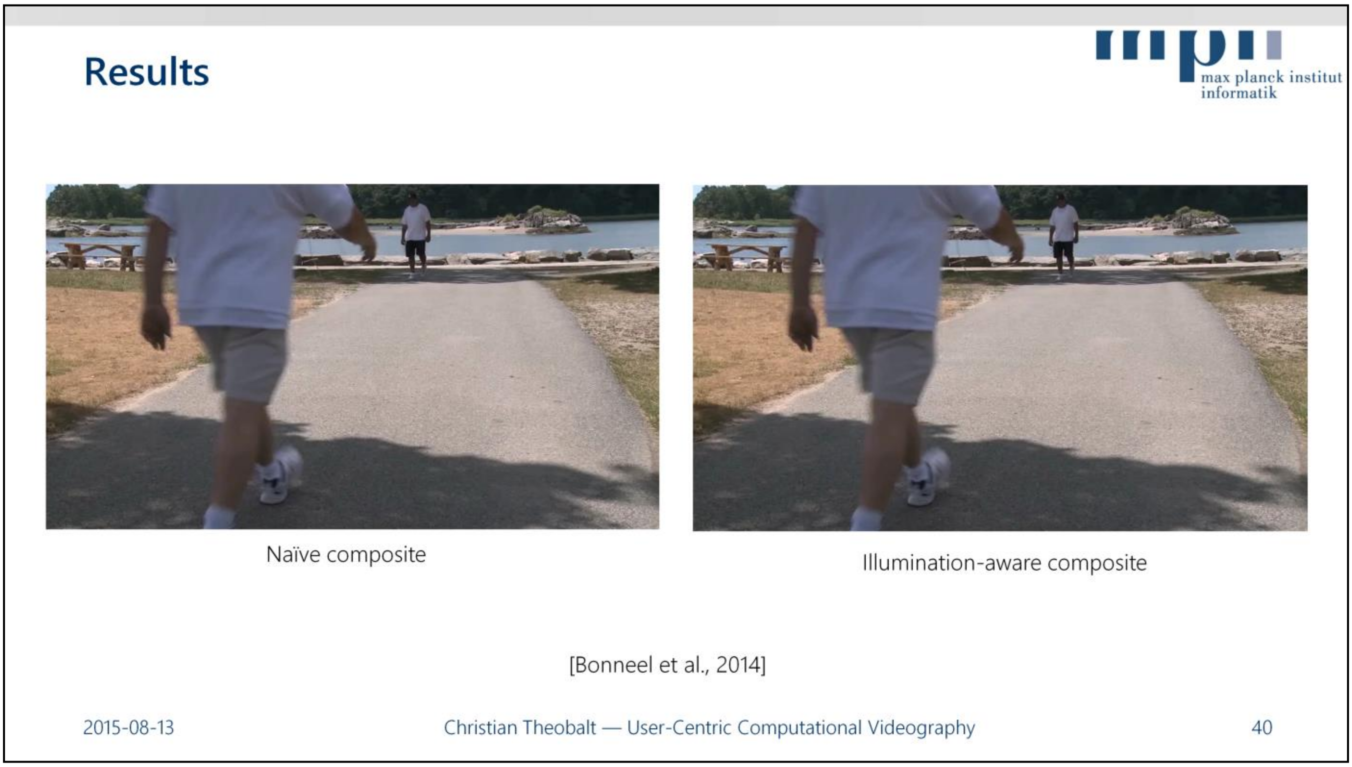
Input

Recoloring

[Bonneel et al., 2014]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 39

This video shows an example of reflectance editing over an entire video. In these results, the reflectance image was painted over in one frame and the modified reflectance was then propagated to the subsequent frames by means of optical flow.



Results

Naïve composite

Illumination-aware composite

[Bonneel et al., 2014]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 40

The video on the left shows a composite in which a segmented region from one video is simply pasted over another video without any illumination adjustment. The video on the right shows an example of lighting consistent video compositing with the approach by Bonneel et al. If S1 and S2 are the shading layers of the videos to be composited, the shading layer in the composited results was set to min(S1,S2) for the segmented foreground region.

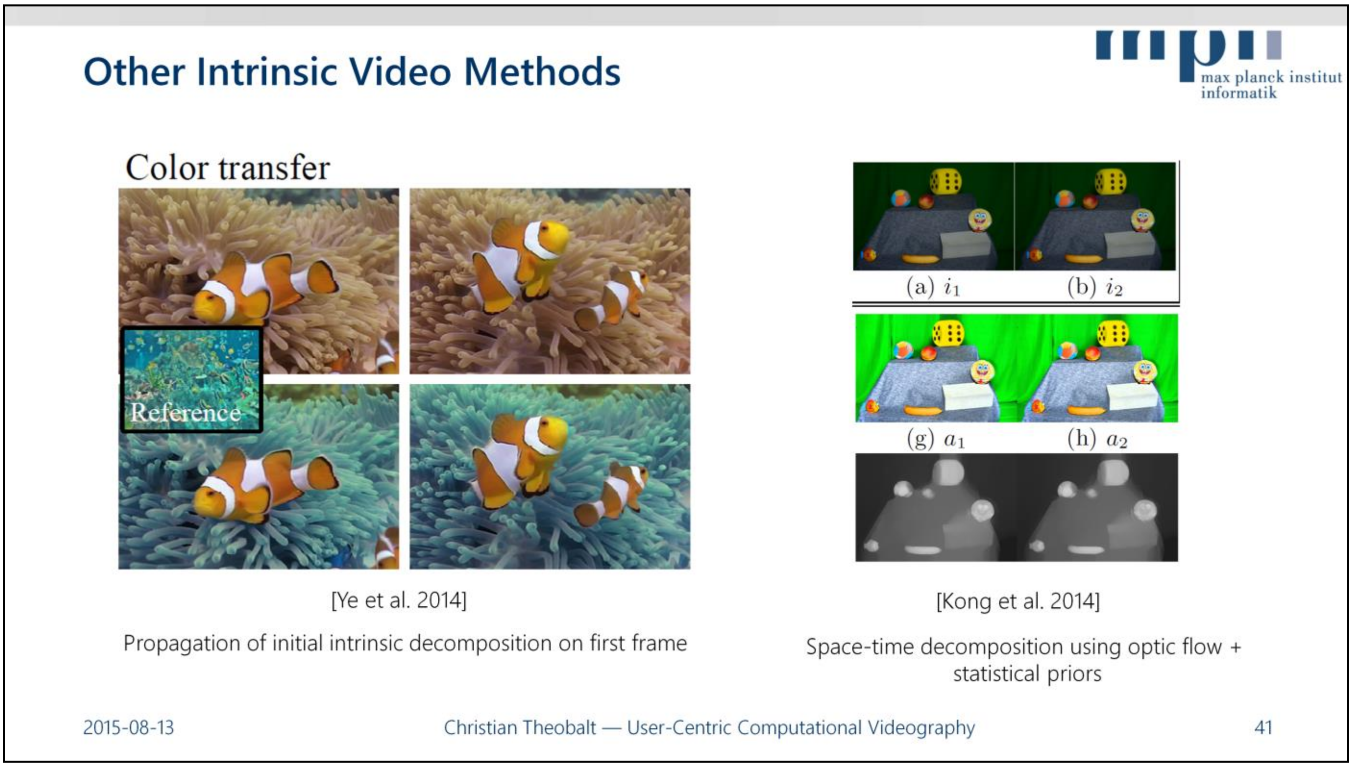
The approach by Bonneel is a step ahead in intrinsic video decomposition, but is still subject to several imitations:

-Since the method assumes monochromatic lighting, the presence of colored lighting or strongly colored interreflections will lead to artefacts in the decomposition.

-For most results shown in their video, around 20 minutes of user interaction was required. The per frame computation times are in the range of 0.2-1.3 seconds for videos of up to 1280x960 resolution; this performance is sufficient for short term edits but may be unsuitable for more complex modification of longer videos where visual feedback may not be computed fast enough.

-Many of the implicitly made assumptions, such as smoothly varying illumination, are violated in real video sequences with high frequency illumination effects, such as hard cast shadows or high frequency geometry; this often leads to erroneous high frequency residuals in reflectance.

-Further on, many limitations of existing approaches are due to the underlying appearance model. Artefacts can be expected around non-diffuse surfaces, and materials with notable sub-surface light transport effects.



Other Intrinsic Video Methods

[Ye et al. 2014]

Propagation of initial intrinsic decomposition on first frame

[Kong et al. 2014]

Space-time decomposition using optic flow + statistical priors

2015-08-13 Christian Theobalt — User-Centric Computational Videography 41

The research on intrinsic video decomposition is still at an early stage. The following works are other examples of recently proposed intrinsic video decomposition approaches, and we briefly look at their core ideas.

The core idea behind the approach presented [Ye et al. 2014] is to compute an intrinsic decomposition on an initial frame and propagate it as good as possible over time. Propagating per-pixel reflectance over time is phrased as a relaxed statistical propagation approach based on a Bayesian framework that finds a Maximum-a-Posteriori (MAP) solution. To avoid accumulation of errors over time, a local confidence threshold is defined and propagation is stopped when the number of unreliable pixels surpasses it. The shading is then leveraged to complete the reflectance layer at the stopping frame, and then information is propagated backwards. This process is iterated in a coarse-to-fine way. The approach also allows for user-guidance with scribbles. Despite appealing decomposition results, the reported computational performance of around 1 minute per frame on a video of 800x600 pixel resolution makes it less suitable for interactive editing applications.

The method described in [Kong et al., 2014] computes reflectance and shading by solving an optimization problem with temporal regularization using optical flow correspondences. A spatial prior on shading is also proposed that is inspired by regularization terms used in previous optical flow approaches. Additionally, some statistical priors on shading are used as proposed by Barron et al. [2015].

Intrinsic video decomposition methods are still at an early stage and many challenges remain unresolved – the high computational complexity makes many approaches computationally prohibitive, and the simplifying assumptions on scene reflectance will need to be relaxed in order to obtain good results on more general scenes.

References Intrinsic Imaging / Video

Intrinsic Imaging:

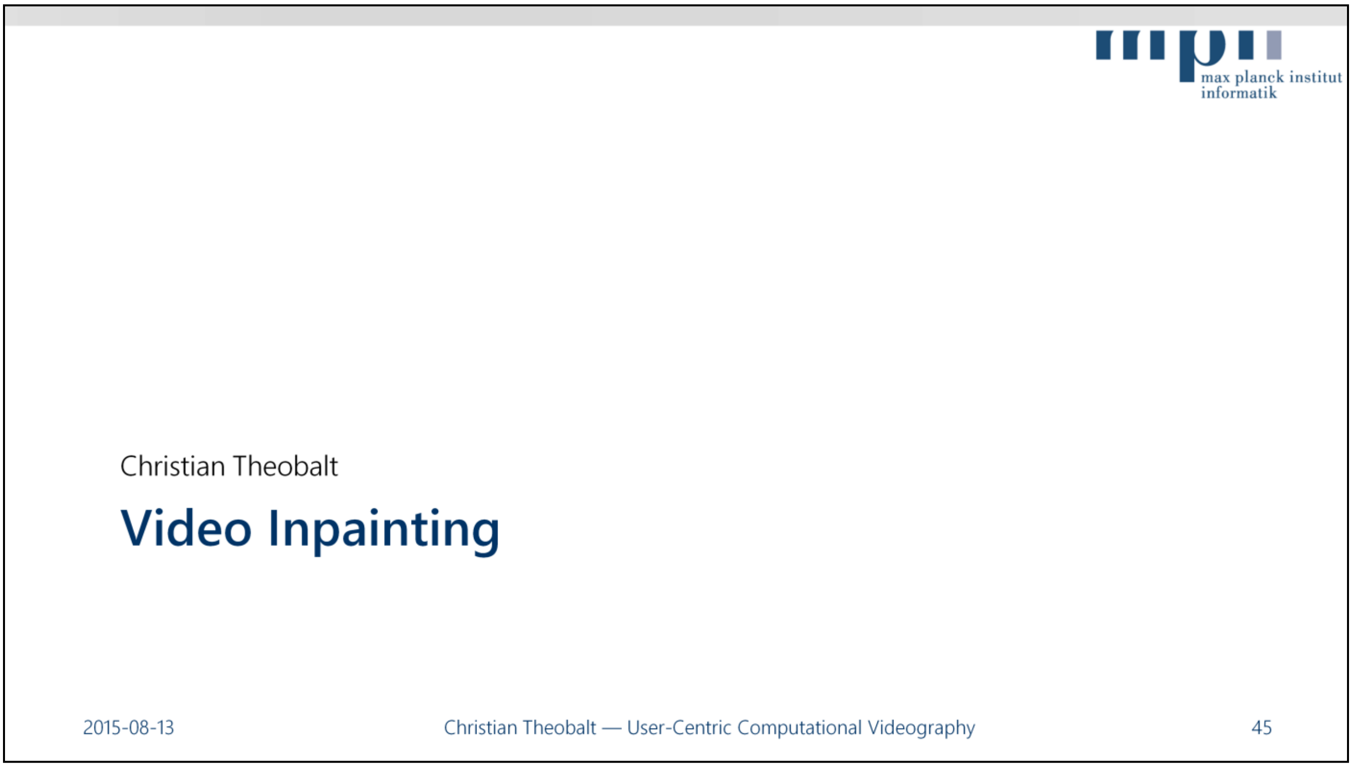
- H. Land and J. J. McCann. Lightness and retinex theory. Journal of the Optical Society of America , 61(1), 1971
- P. Sinha and E. Adelson. Recovering reflectance and illumination in a world of painted polyhedra. In ICCV, pages 156–163, 1993.
- M. Bell and W. T. Freeman. Learning local evidence for shading and reflectance. In Proc. ICCV, volume 1, pages 670–677, 2001
- M. F. Tappen, E. H. Adelson, and W. T. Freeman. Estimating intrinsic component images using non-linear regression. In Proc. CVPR, volume 2, pages 1992–1999, 2006.
- M. F. Tappen, W. T. Freeman, and E. H. Adelson. Recovering intrinsic images from a single image. IEEE Trans. on Pattern, Analysis and Machine Intelligence, 27(9):1459–1472, 2005.
- L. Shen, P. Tan, and S. Lin. Intrinsic image decomposition with non-local texture cues. In Proc. CVPR. pages 1–7, 2008
- Y. Weiss. Deriving intrinsic images from image sequences. In Proc. ICCV, volume 2, pages 68–75, 2001.
- Y. Matsushita, K. Nishino, K. Ikeuchi, and M. Sakauchi. Illumination normalization with time-dependent intrinsic images for video surveillance. In Proc. CVPR, volume 1, pages 3–10, 2003.
- Y. Matsushita, S. Lin, S. B. Kang, and H.-Y. Shum. Estimating intrinsic images from image sequences. In Proc. ECCV, volume 2, pages 274–286, 2004.
- R. Grosse, M. K. Johnson, E. H. Adelson, and W. T. Free-man. Ground truth dataset and baseline evaluations for intrinsic image algorithms. In ICCV 2009.

References Intrinsic Imaging / Video

- Chen, Q. and Koltun, V., A Simple Model for Intrinsic Image Decomposition with Depth Cues, ICCV 2013.
- R. Kimmel, M. Elad, D. Shaked, R. Keshet, and I. Sobel. A variational framework for retinex. IJCV, 52:7–23, 2003.
- P. Y. Laffont, A. Bousseau, S. Paris, F. Durand, and G. Drettakis. 2012. Coherent intrinsic images from photo collections. ACM TOG , 31, 6, 2012.
- P. Y. Laffont, A. Bousseau, G. Drettakis, IEEE TVCG, Rich Intrinsic Image Decomposition of Outdoor Scenes From Multiple Views, 2013.
- L. Shen and C. Yeo. Intrinsic images decomposition using a local and global sparse representation of reflectance. In CVPR, 2011.
- P. V. Gehler, C. Rother, M. Kiefel, L. Zhang, and B. Scholkopf. Recovering intrinsic images with a global sparsity prior on reflectance. In NIPS, 2011.
- J.T. Barron, J. Malik, Shape, Illumination, and Reflectance from Shading, IEEE Trans. PAMI, 2015.
- J.T. Barron, J. Malik , Intrinsic Scene Properties from a Single RGB-D Image , Proc. CVPR 2013.
- S. Bell, K. Bala, and N. Snavely. 2014. Intrinsic images in the wild. ACM TOG 33, 4, Article 159, 2014.
- OMER, I., AND WERMAN, M. 2004. Color lines: Image specific color representation. In Proc. CVPR.
- HSU, E., MERTENS, T., PARIS, S., AVIDAN, S., AND DURAND, F., Light mixture estimation for spatially varying white balance. ACM Trans. Graph. (Proc. SIGGRAPH), 2008.

References Intrinsic Imaging / Video

- Barrow, H.G., Tenenbaum, J.M.: Recovering intrinsic scene characteristics from images. In: Computer Vision Systems. pp. 3{26 (1978)
- K. J. Lee, Q. Zhao, X. Tong, M. Gong, S. Izadi, S. U. Lee, P. Tan, and S. Lin. Estimation of intrinsic image sequences from image+depth video. In ECCV 2012.
- G. Ye, E. Garces, Y. Liu, Q. Dai, and D. Gutierrez. 2014. Intrinsic video and applications. ACM Trans. Graph. 33, 2014.
- N. Bonneel, K. Sunkavalli, J. Tompkin, D. Sun, S. Paris, and H. Pfister. Interactive intrinsic video editing. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 33, 6, 2014.
- BOUSSEAU, A., PARIS, S., AND DURAND, F. User-assisted intrinsic images. ACM Trans. Graph. (Proc. SIGGRAPH) 28, 5, 2009.



Christian Theobalt

Video Inpainting

2015-08-13 Christian Theobalt — User-Centric Computational Videography 45

In the following, we will look into another video editing operation that is often required in conjunction with other edits, such as removal of an object from video, namely video inpainting.

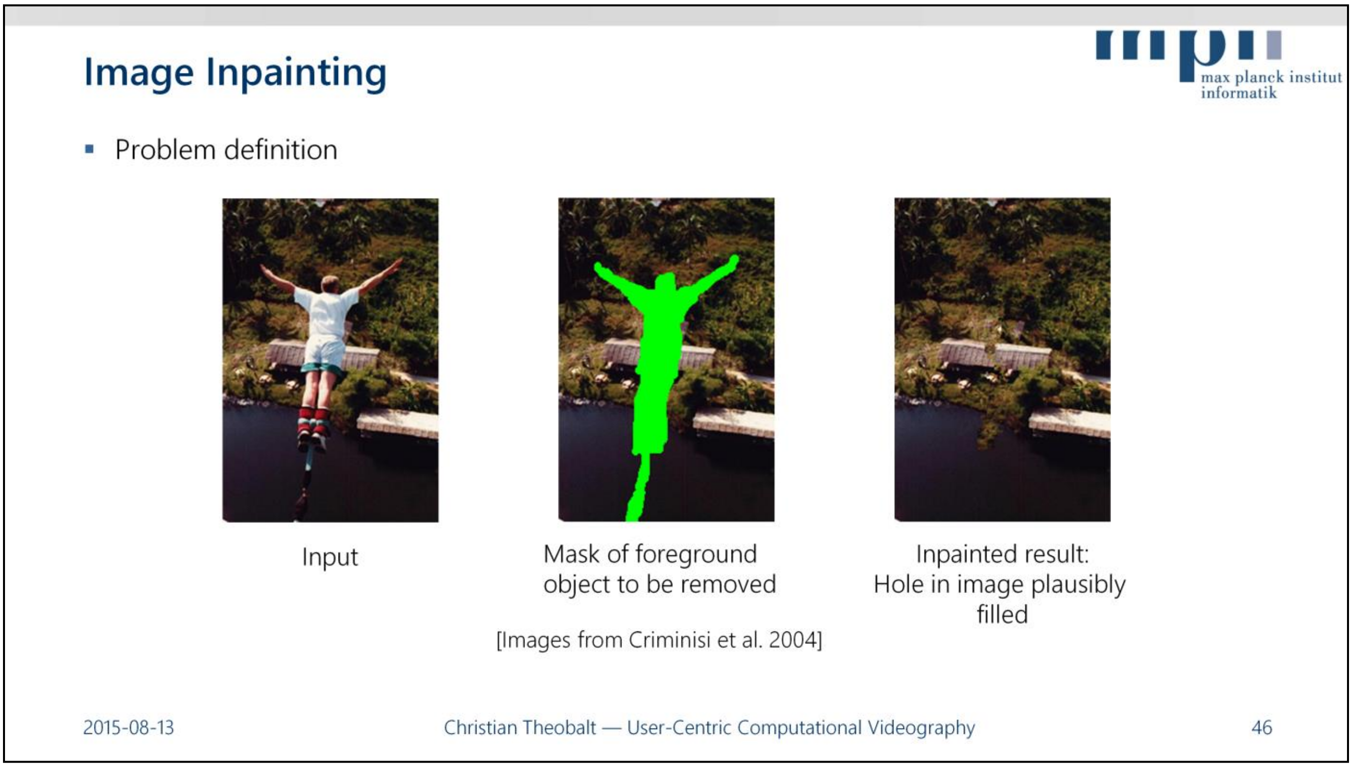


Image Inpainting

- Problem definition

Input

Mask of foreground object to be removed

Inpainted result: Hole in image plausibly filled

[Images from Criminisi et al. 2004]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 46

Before looking into video inpainting algorithms, let us look again into the related domain of image inpainting to define the problem. In image inpainting, the goal is to algorithmically fill in a designated image region with plausible content such that the edit is visually imperceptible. A typical use case for this technique is the filling of a hole after removal of a certain pixel region, as shown in the example above. A variety of image-based algorithms to solve this problem have been proposed in the past, and in the following we will look into important categories. Many of their core ideas have inspired subsequent work on video inpainting.

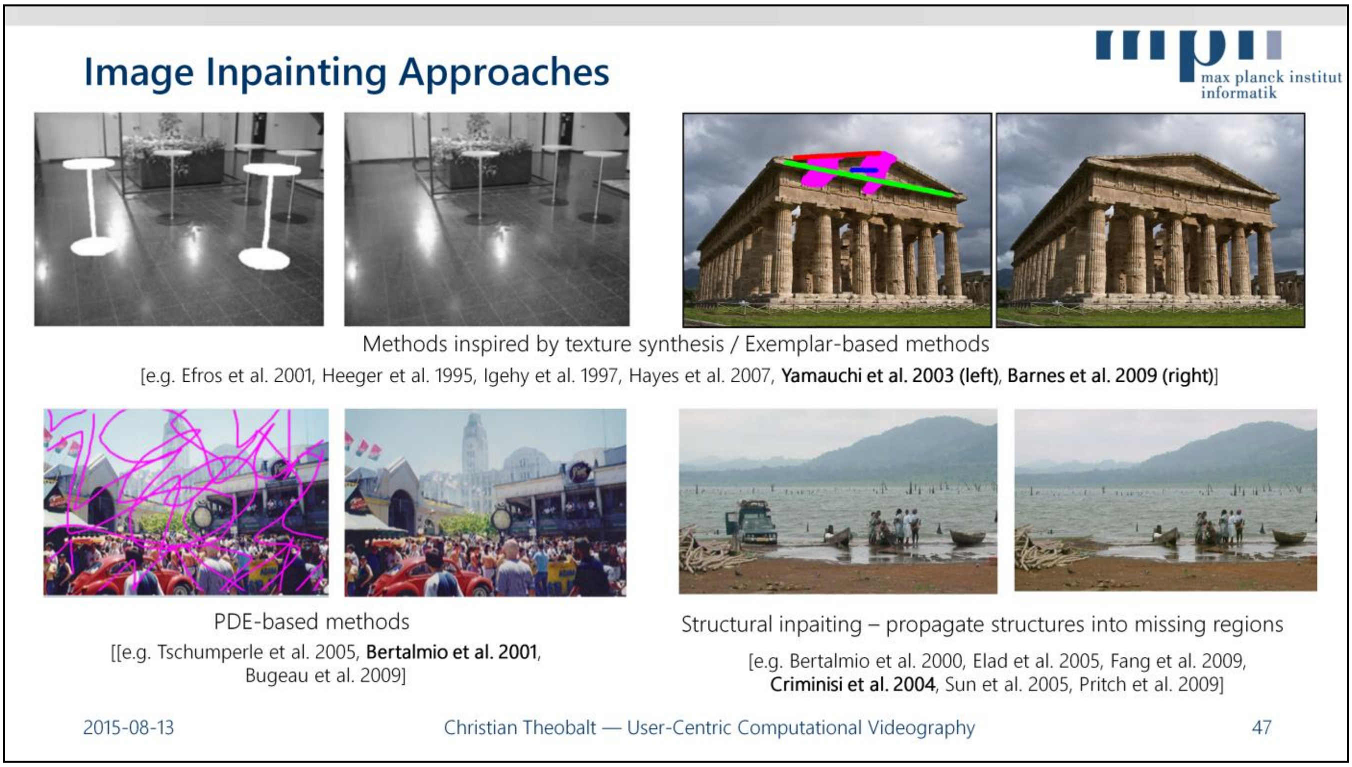


Image Inpainting Approaches

Methods inspired by texture synthesis / Exemplar-based methods

[e.g. Efros et al. 2001, Heeger et al. 1995, Igehy et al. 1997, Hayes et al. 2007, **Yamauchi et al. 2003 (left)**, **Barnes et al. 2009 (right)**]

PDE-based methods

[[e.g. Tschumperle et al. 2005, **Bertalmio et al. 2001**, Bugeau et al. 2009]

Structural inpaiting – propagate structures into missing regions

[e.g. Bertalmio et al. 2000, Elad et al. 2005, Fang et al. 2009, **Criminisi et al. 2004**, Sun et al. 2005, Pritch et al. 2009]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 47

In the past, computer vision and computer graphics researchers have proposed a variety of approaches to inpaint holes in a single image. In the following, we list example works from important categories of approaches; an exhaustive review of works in this specific domain is beyond the scope of the course.

Several methods are inspired by the concept of texture synthesis. Essentially, the idea is to create new patches to be filled into the how from “exemplar” patches in the surrounding areas of the image [e.g. Efros et al. 2001, Heeger et al. 1995, Igehy et al. 1997, Yamauchi et la. 2003]. Algorithms differ in the way how they preserve continuity between pixels filled into the hole and original image pixels, as well as what additional transformations on patches (e.g. appearance) are considered and allowed. Exemplar-based approaches typically require an expensive search step to find most similar / suitable patches in other image regions for hole completion.

The shift map algorithm by Pritch et al. is an example from that category. It can be used for a variety of image editing tasks, and for inpaiting it optimizes a vector field of displacements for pixel in the howl that indicates from where content should be copied. Several video inpainting approaches are inspired by it. Searching for most similar pixels / patches is a computationally expensive operation. The PatchMatch approach by Barnes et al. [2009] proposes a highly effective randomized search method for nearest neighbor patches, and it can be used in several image editing tasks, including hole filling.

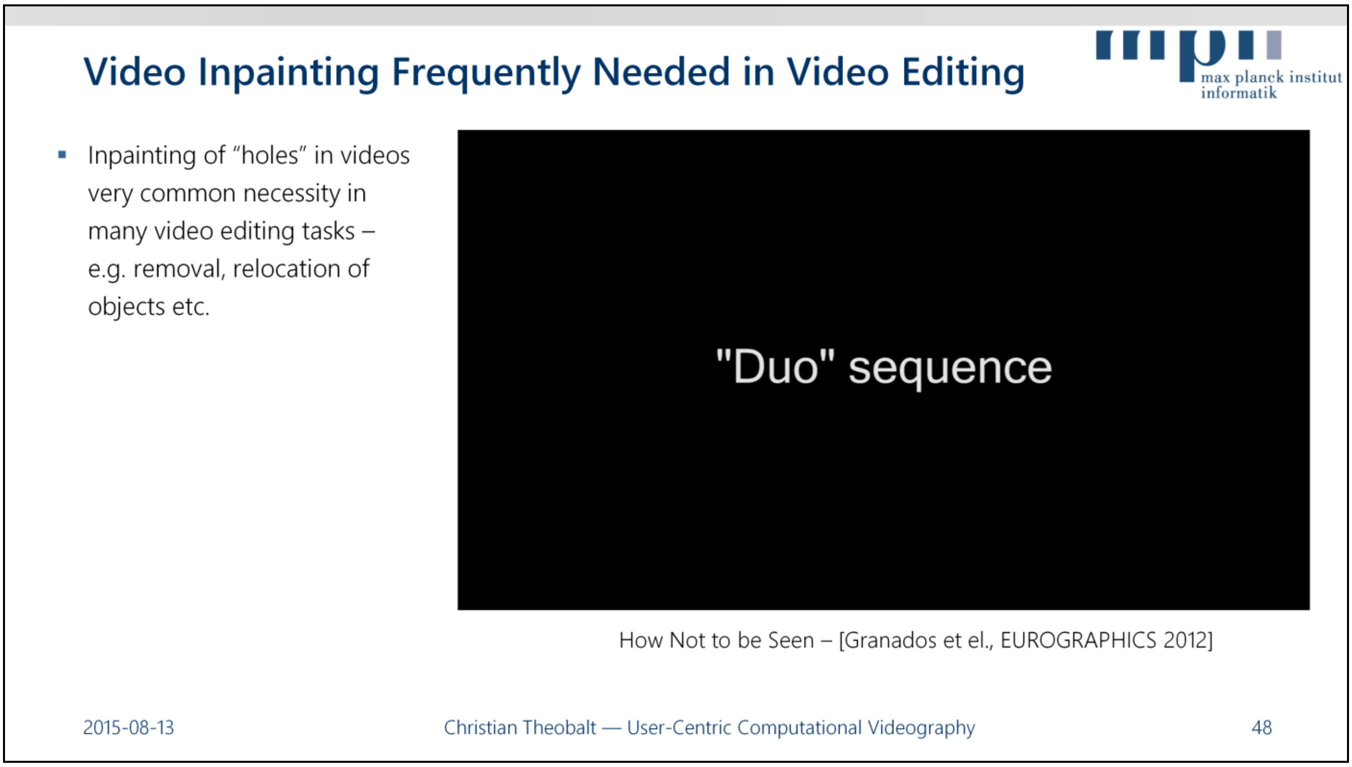
Structural inpainting approaches combine the benefits of texture synthesis approaches with the explicit propagation of structural elements seen in the surroundings of a hole to inside the hole [e.g. Bertalmio et al. 2000, Elad et al. 2005, Fang et al. 2009]. For instance, Criminisi et al. [2004]

combine exemplar-based synthesis with isophote propagation into the hole, and patch-based inpaiting around user-specified structures is presented in [Sun et al. 2005]. An exemplar-based approach that exploits community photo collections as database and local context similarity is presented in [Hayes et al., 2007].

Another category of approaches models image inpainting as a diffusion process that propagates information from the surroundings into the hole [e.g. Bertalmio et al. 2000, Oliveira et al. 2001]. The final solution is found by solving a partial differential equation [e.g. Tschumperle et al. 2005]. Bertalmio et al. [2001] propose a PDE-based method inspired by fluid dynamics. The effective combination of texture synthesis and diffusion has also been investigated [Bugeau et al. 2009].

As with several previously studied effects, naively applying image inpainting approaches to every frame of video will usually not lead to satisfactory results. The inpainted frames will not be temporally coherent. Therefore, specific video inpainting approaches were designed. The video inpainting problem is further complicated by the dynamics of video – while inpainting a static background region in

a video filmed with a static camera is most similar in setting to image inpaiting, in video one may need to inpaint dynamic region sin the background, and the camera may in addition move from frame to frame. This severely complicates the search for space-time coherent content to be inserted into the hole.



Video Inpainting Frequently Needed in Video Editing

- Inpainting of “holes” in videos very common necessity in many video editing tasks – e.g. removal, relocation of objects etc.

"Duo" sequence

How Not to be Seen – [Granados et el., EUROGRAPHICS 2012]

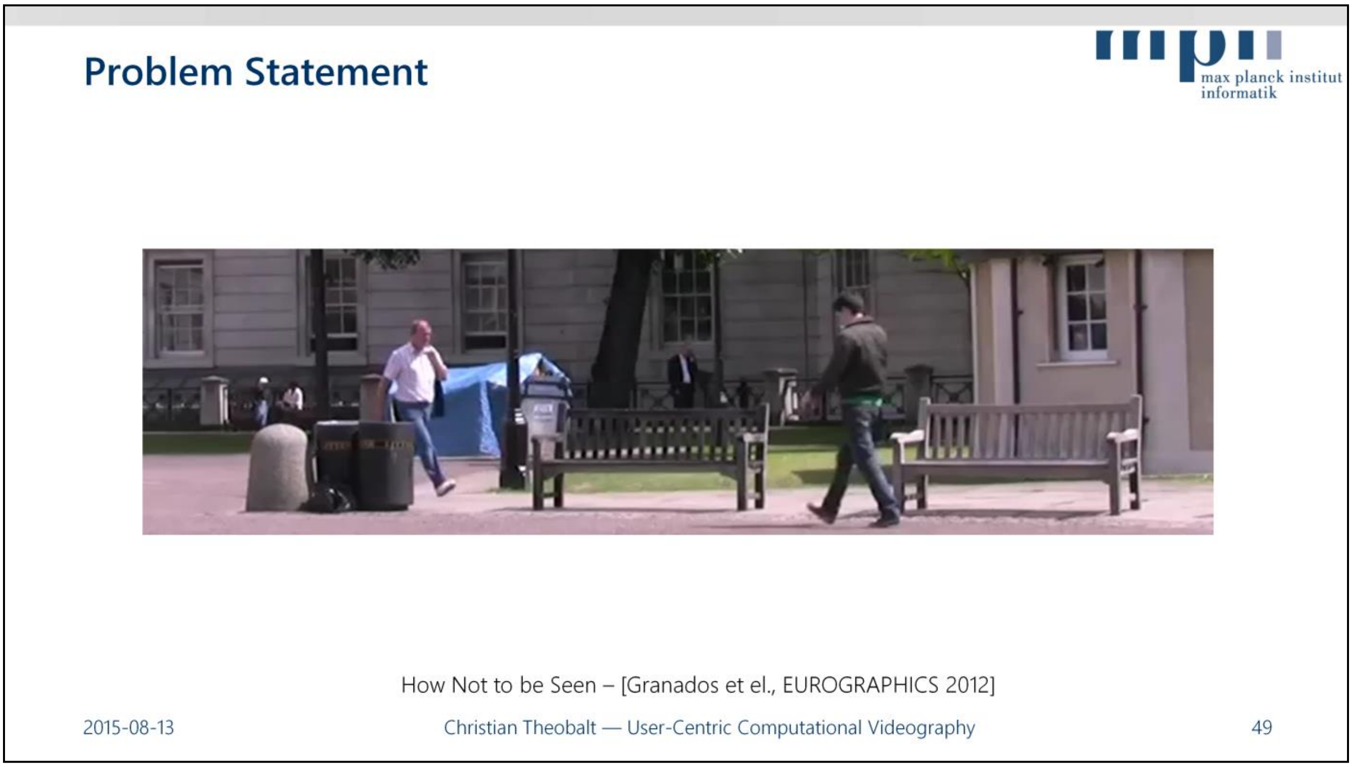
2015-08-13 Christian Theobalt — User-Centric Computational Videography 48

Often times, one would like to remove an object from video, for instance because during video recording a dynamic foreground object may have erroneously occluded the action which one actually wanted to capture. An example of this is shown in the video above, where some persons in the foreground erroneously occluded the band in the background. A video inpainting approach would need to fill in the dynamic background, after the people in the foreground where removed. The video shows the result created automatically by the video inpainting approach of Granados et al. [2012]. Please note that the particular difficulty here is that the dynamic background needs to be inpainted.

In addition to cases where the actual goal is to remove objects from a video, the need to inpaint holes in the background may arise in conjunction with other video edits, such as the spatio-temporal relocalization of an object, the change of motion of the object, or the change of appearance of certain elements of the video; we will see further examples of this in the later part of the course on model-based video editing.

In the following, we will briefly define the video inpainting problem, identify its main challenges, and discuss in more detail the solution proposed in the paper by Granados et al. from EUROGRAPHICS 2012 as one example of a recent video inpaiting approach.

Video link: http://gvv.mpi-inf.mpg.de/projects/vidinp/



Problem Statement

How Not to be Seen – [Granados et el., EUROGRAPHICS 2012]

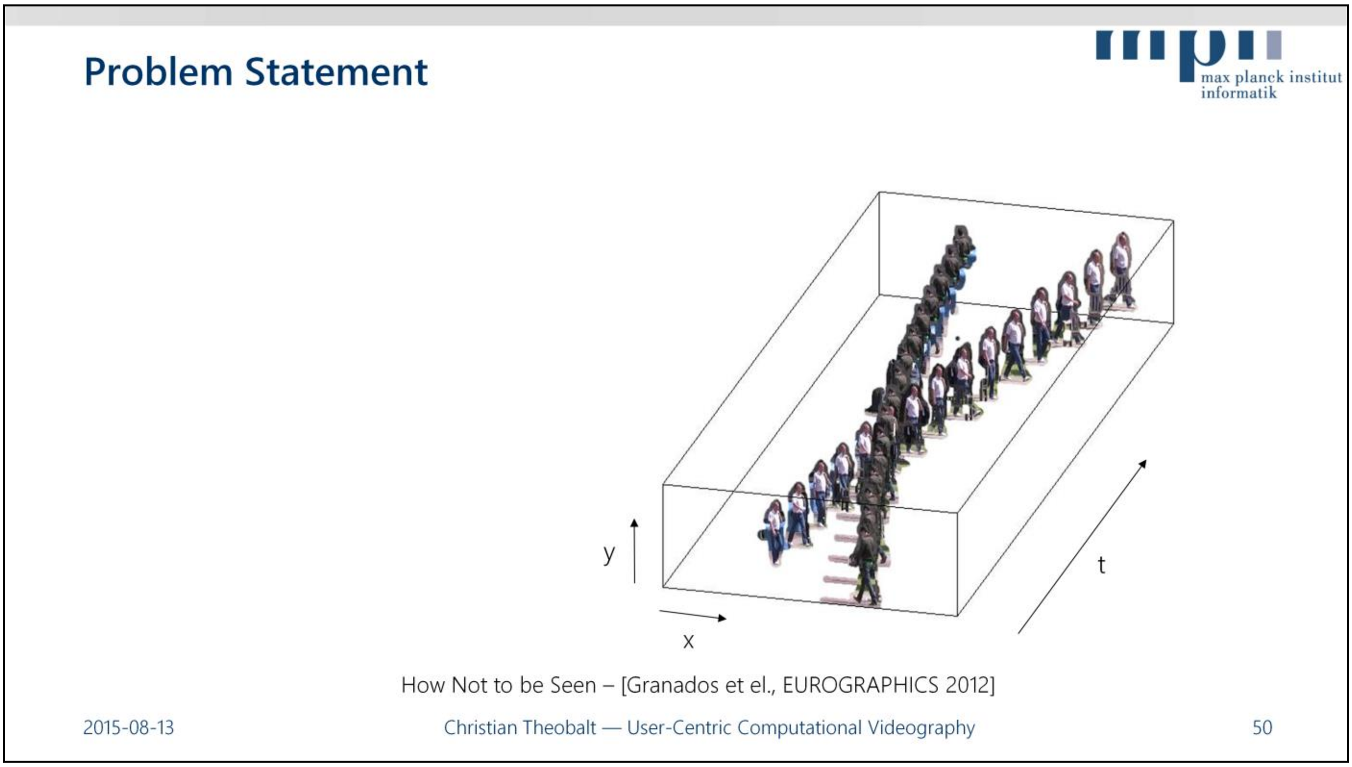
2015-08-13 Christian Theobalt — User-Centric Computational Videography 49

In order to define the video inpainting problem, identify its main challenges, and in order to see what specific scenario the approach by Granados et al. [EG 2012] takes, let’s have a look at the following video. In this video, the two persons walk in opposite directions, and we see that one occludes the other at some point.

As stated before, a typical case where video inpainting is needed is if one wants to remove one of the persons from the video. To remove the person in the back, with the white shirt, is comparably easy since it covers just the static background. In that case one can find the content to inpaint into the background in other frames showing the same unoccluded X/Y region. In particular if the camera is not moving, this is relatively straightforward. In such a case, applying a filter to all the pixel values over time (e.g. median filter) often already produces a very satisfactory background inpainting result.

It would be more difficult to remove the person in front, because he covers the moving person behind, and that missing region that needs to be inpainted is dynamic, and thus much more challenging to restore.

Video link: http://gvv.mpi-inf.mpg.de/projects/vidinp/



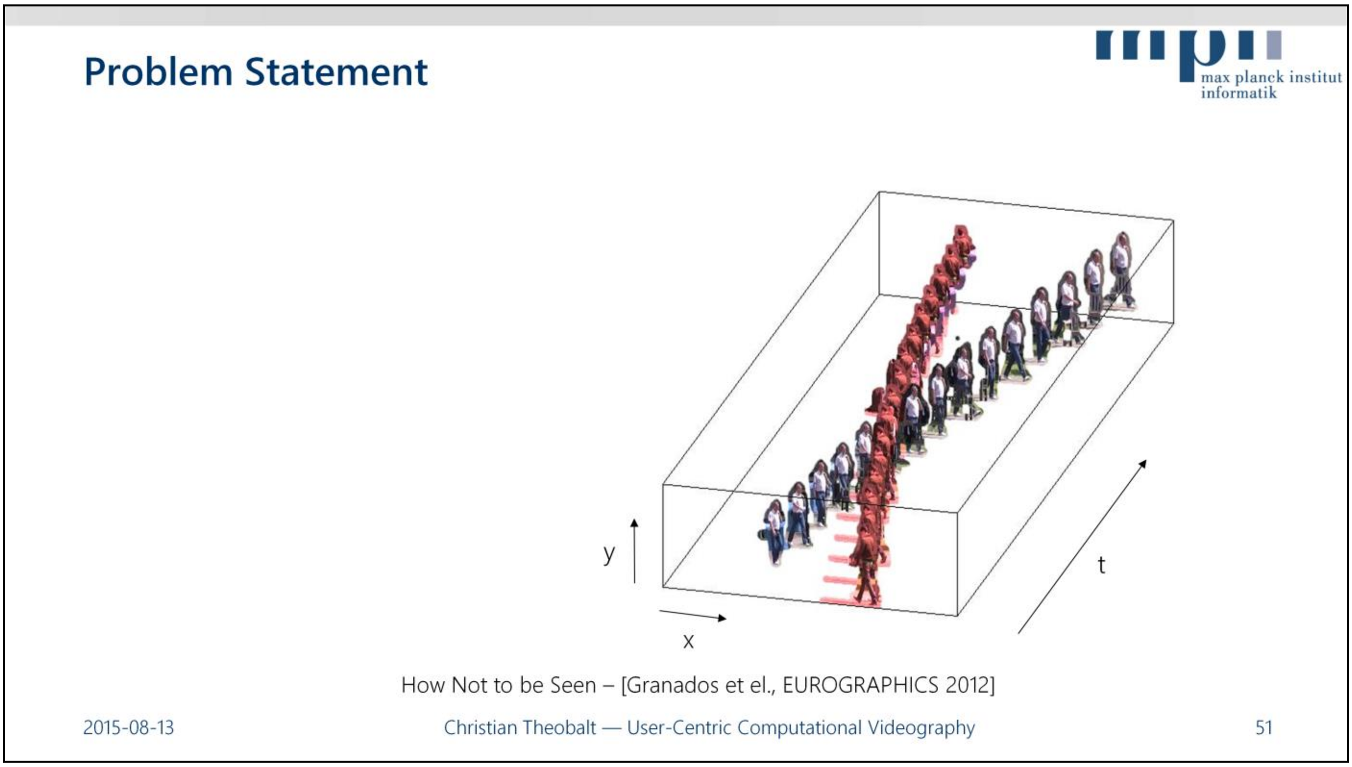
Problem Statement

max planck institut
informatik

How Not to be Seen – [Granados et al., EUROGRAPHICS 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 50

This is a visualization of the same video as a video volume. In this volume, the frames are stacked one after the other in the order they appear in the video. In this figure, we don't show the background for better visualization.

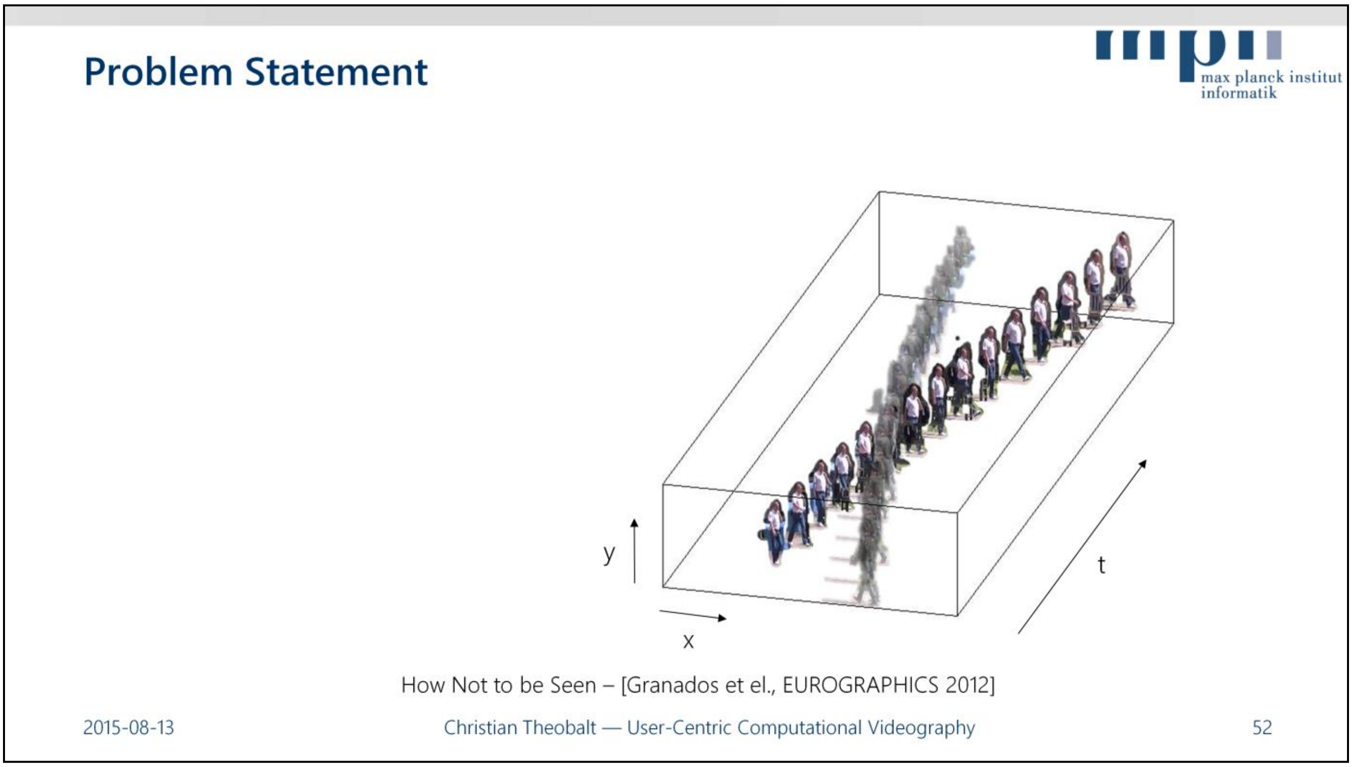


Problem Statement

max planck institut informatik

How Not to be Seen – [Granados et el., EUROGRAPHICS 2012]

This region in red is what the user wants us to remove.

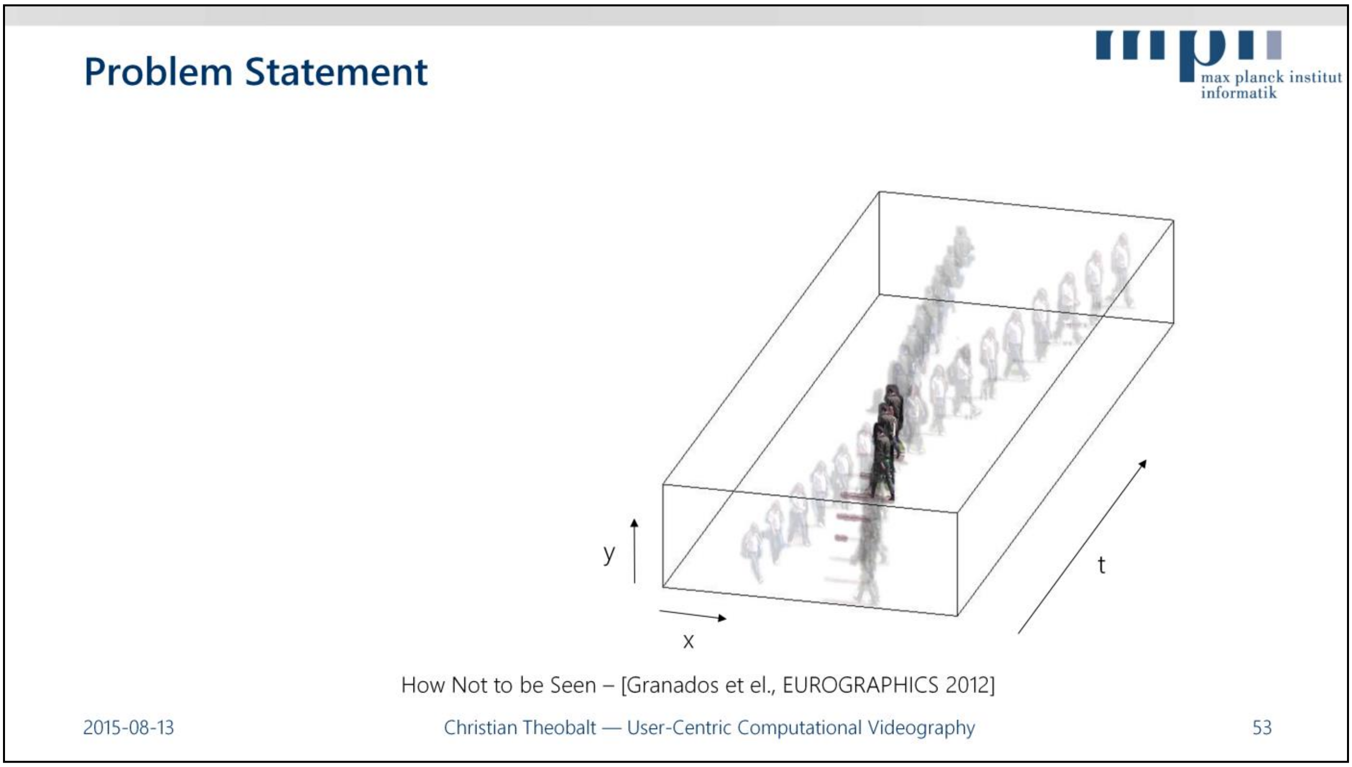


Problem Statement

How Not to be Seen – [Granados et el., EUROGRAPHICS 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 52

This means that one has to restore the scene content occluded by this person. The easier case is inpainting all those parts of the video where the person only occludes static background. In that case one can find the content to inpaint into the background in other frames showing the same unoccluded X/Y region. In particular if the camera is not moving, this is relatively straightforward, since corresponding regions over time stay at the same X/Y location. In such a case, applying a filter to all the pixel values over time (e.g. median filter) often already produces a very satisfactory background inpainting result.



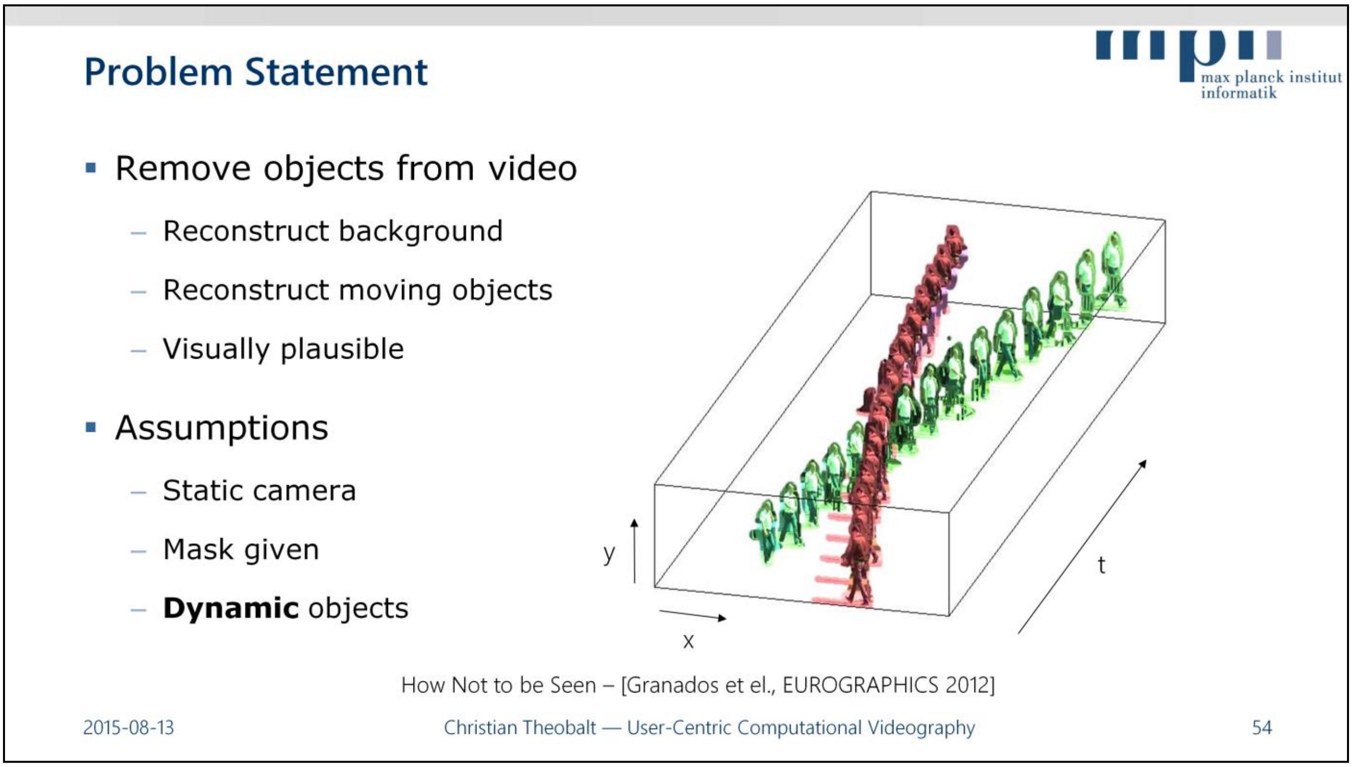
Problem Statement

How Not to be Seen – [Granados et al., EUROGRAPHICS 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 53

The harder case is inpainting those regions where the other person was covered. The problem is that the region was not static during the occlusion. To restore this moving background is much more difficult. It is not enough to search for other (spatio-)temporal locations in the video where the person was visible, one also needs to find and transfer these regions in such a way that the dynamics of the scene in the target region is plausibly reproduced.

The problem can be slightly simplified if one assumes a parametric model of the occluded region, e.g. one assumes there is a moving person and thus models the motion in the region to be inpainted with some human template (see discussion of alternative video inpaiting methods at the end of this section). However, this also limits the generality of the approach. The approach by Granados et al. [EG 2012] does not use such higher-level shape or motion priors and is purely based on searching similar space-time regions in unoccluded regions of the video volume, as we will explain later.



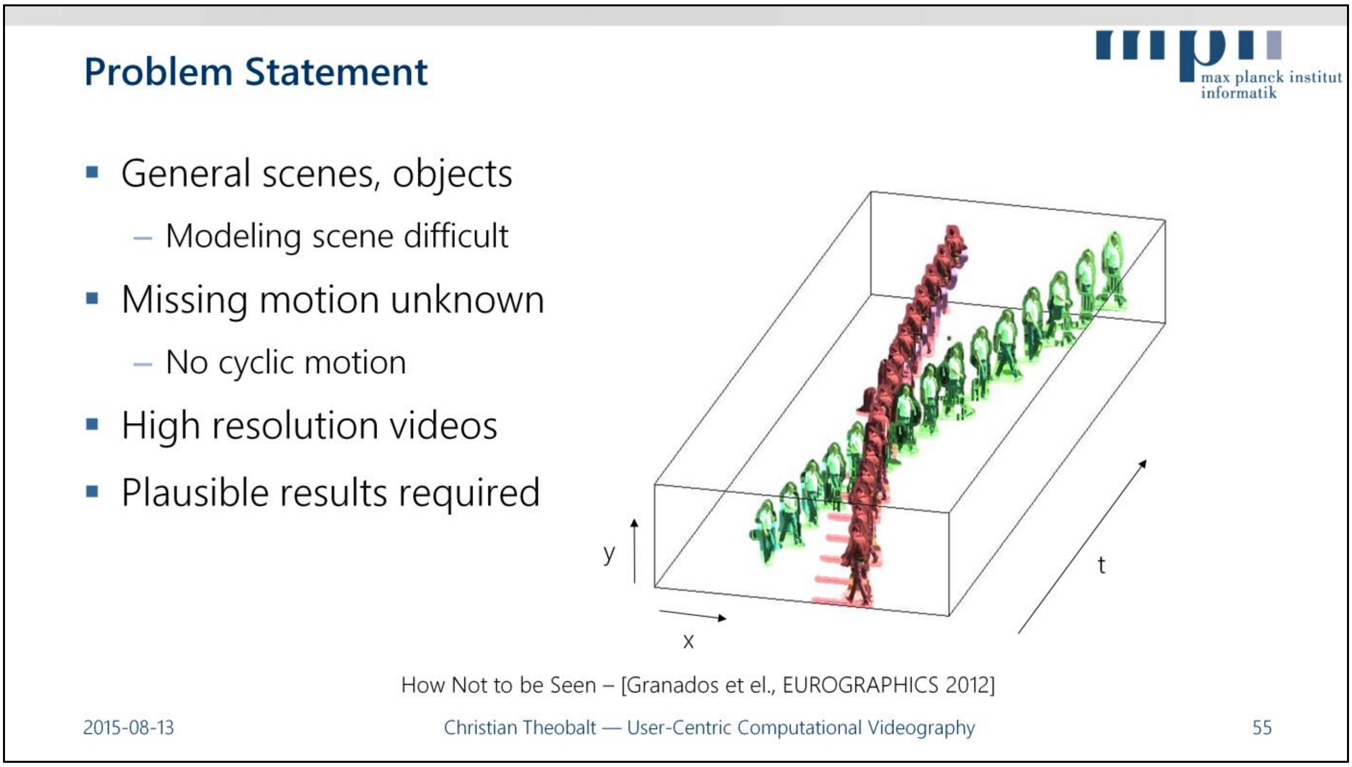
Problem Statement

- Remove objects from video
  - Reconstruct background
  - Reconstruct moving objects
  - Visually plausible
- Assumptions
  - Static camera
  - Mask given
  - **Dynamic** objects

How Not to be Seen – [Granados et el., EUROGRAPHICS 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 54

To summarize, if one removes an object from video, one needs to reconstruct the scene behind, including the background and all moving objects, in a visually plausible way. In order to solve this problem, the method by Granados et al. [EG 2012] makes a few assumptions: It is assumed that the camera is static, and it is assumed that a mask for the object to be removed is given (e.g. using video segmentation approaches discussed earlier in this course). Further on, the algorithm approaches the general case where the region to be removed and the background to be inpainted can be non-static.



Problem Statement

max planck institut informatik

- General scenes, objects
  - Modeling scene difficult
- Missing motion unknown
  - No cyclic motion
- High resolution videos
- Plausible results required

How Not to be Seen – [Granados et el., EUROGRAPHICS 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 55

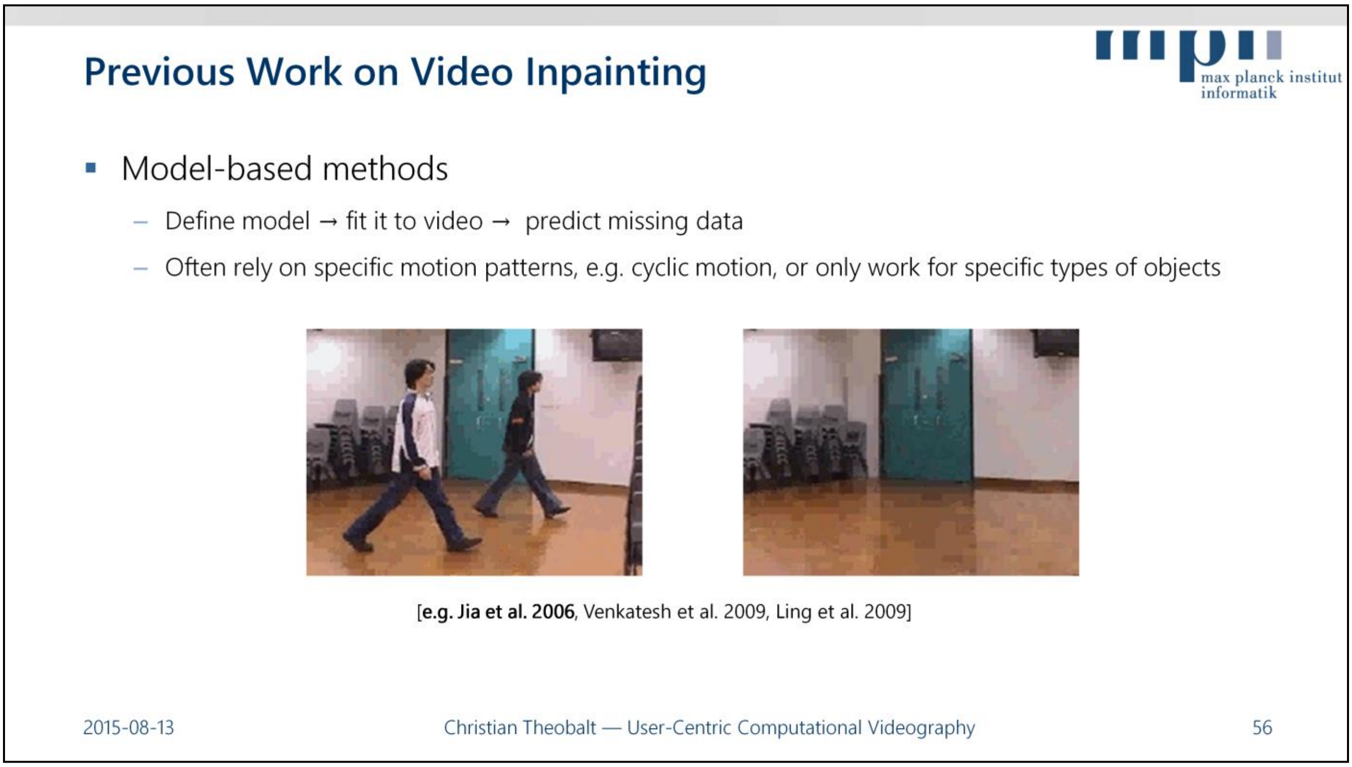
The main challenges are that:

1) The method needs to handle general scenes, where one does not have a model of the scene that could be use to predict the missing data.

2) Also, the missing motion is of course unknown, and one cannot assume that, for instance, it will always be cyclic.

3) And, for being practically useful the algorithm needs to work with high resolution videos, so algorithmic complexity is an issue and computation times need to stay in reasonable bounds.

4) Despite all these challenges, of course, producing visually plausible results is always the ultimate goal.



Previous Work on Video Inpainting

- Model-based methods
  - Define model → fit it to video → predict missing data
  - Often rely on specific motion patterns, e.g. cyclic motion, or only work for specific types of objects

[**e.g. Jia et al. 2006**, Venkatesh et al. 2009, Ling et al. 2009]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 56

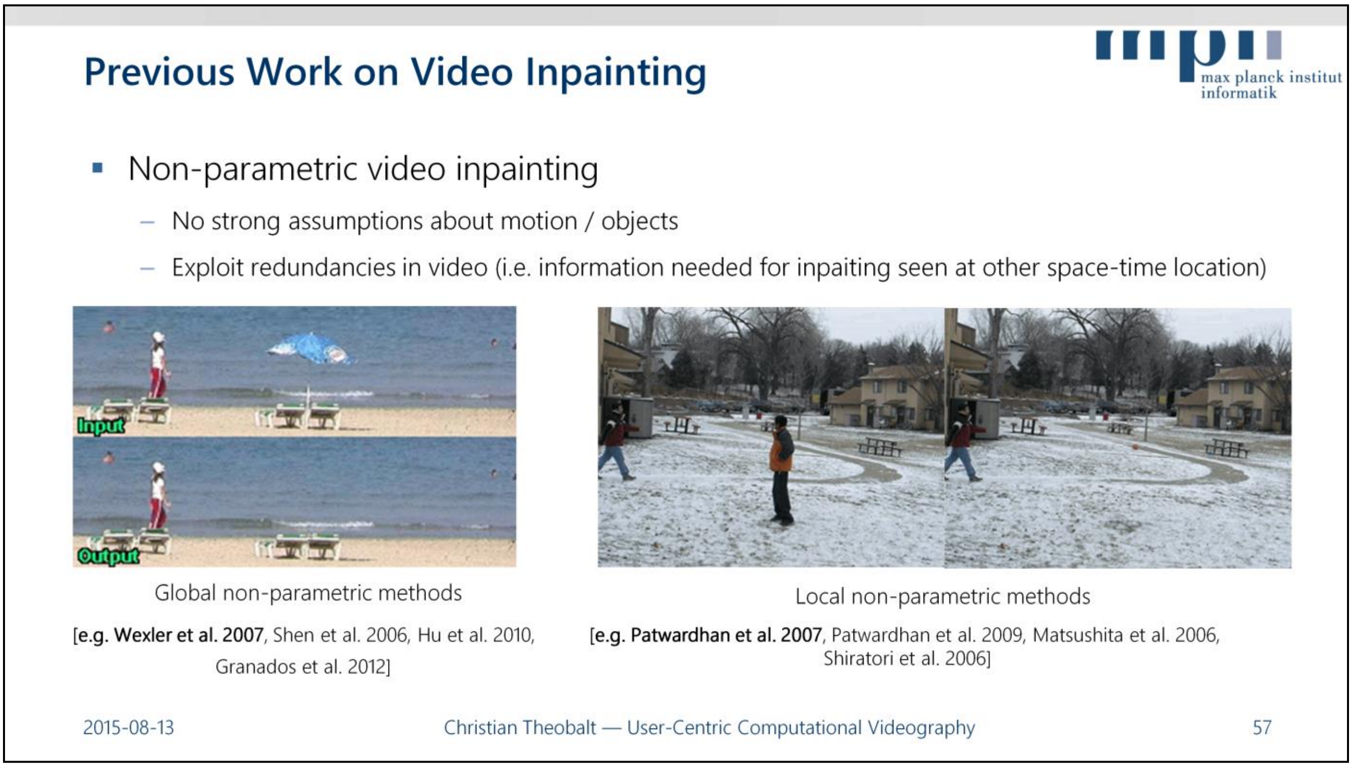
Before we look into the details of the approach by Granados et al., let us first look into some related work on video inpainting. Model-based methods for video inpaiting assume that the behavior of the occluded objects can be somehow modelled, for instance by a shape template or a parametric motion model.

Many approaches thus build and adapt a model for the object, train it with the input video, and then try to predict how the occluded part looks like by using the parametric model. For instance, Venkatesh et al. [2009] track and segment the occluded object. They construct a database of segmented frames where the occluded object is fully visible. Using dynamic programming, the holes are completed by aligning frames in the database to the partially or fully-occluded frames in the hole.

This idea is extended by Ling et al. [2009]. The contours of the object of interest are estimated by using motion information. The contours are then used to retrieve the object frames from a database of unoccluded views, even under some restricted posture change.

Jia et al. [2006] assume a periodic motion for the occluded objects. After segmenting them, the completion problem is cast as a warping and alignment of the object’s visible trajectory with that of its occluded views.

Object-based systems demonstrate plausible completions for certain categories of moving objects, especially humans. However, they either impose an explicit class of possible motions (e.g., cyclic) [Venkatesh et al. 2009, Jia et al. 2006], or require the motion to be simple such that a dense sampling of postures is feasible [Ling et al. 2009].



Previous Work on Video Inpainting

- Non-parametric video inpainting
  - No strong assumptions about motion / objects
  - Exploit redundancies in video (i.e. information needed for inpaiting seen at other space-time location)

Global non-parametric methods

[e.g. **Wexler et al. 2007**, Shen et al. 2006, Hu et al. 2010, Granados et al. 2012]

Local non-parametric methods

[e.g. **Patwardhan et al. 2007**, Patwardhan et al. 2009, Matsushita et al. 2006, Shiratori et al. 2006]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 57

A widely used category of video inpainting methods are non-parametric methods. These methods do not make any strong prior assumptions about the scene or its motion, only that there is enough redundancy so that the missing data can be reconstructed from other parts of the video.

One can further categorize two main classes of non-parametric approaches. The first type of approach completes holes in a video by formulating the problem as an energy minimization problem that is solved globally on the entire hole. Usually, the energy functional enforces similarity between the content of the hole and its surrounding regions (spatially and temporally), as well as coherence criteria for the content filled into the hole. The optimal solution thus describes which content to take from where in the video volume such that these consistency criteria are met.

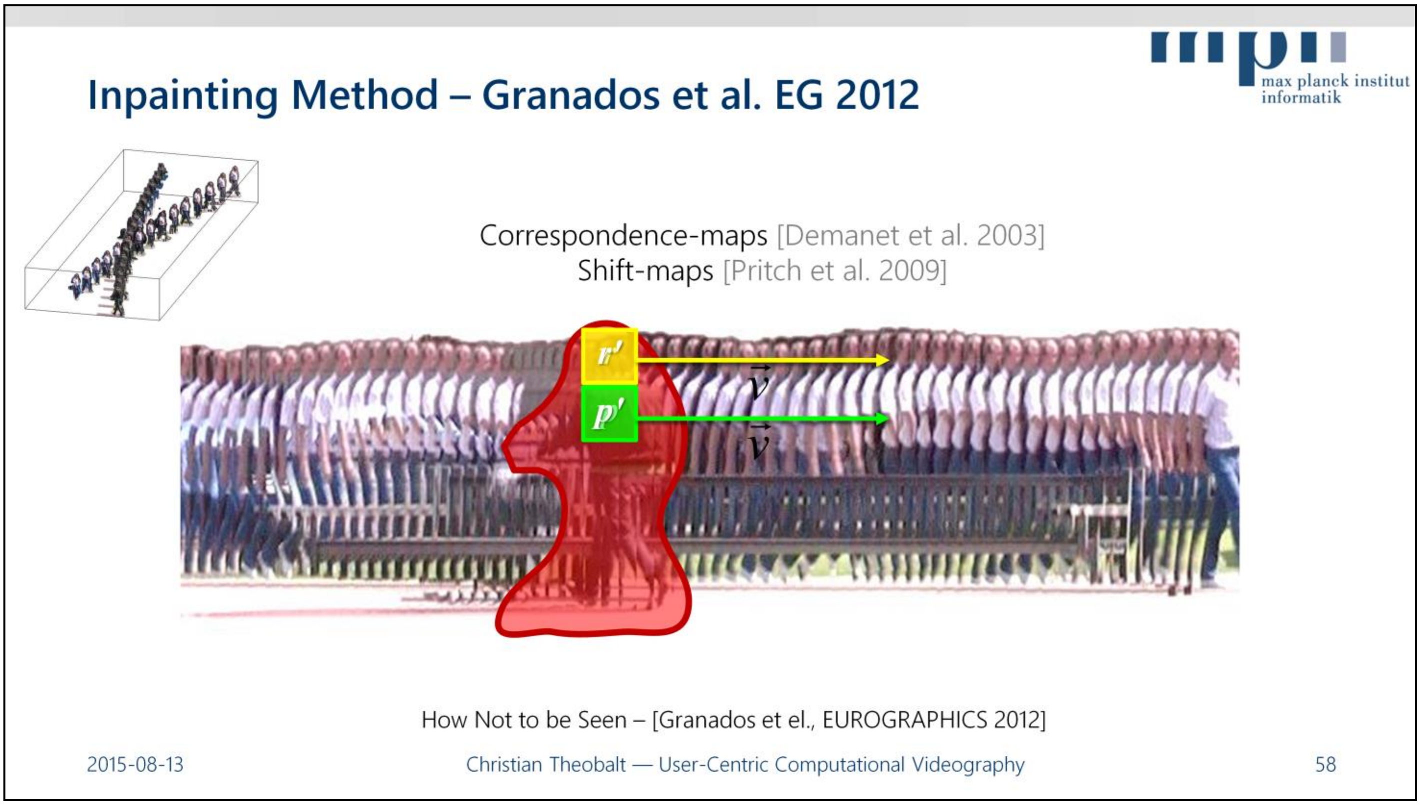
An example for such a global approach is the method by Wexler et al. [2007]. A spatio-temporal patch is sampled from every observed pixel n the input video. The collection of these patches constitutes a database reflecting the statistics of the video. Completion is performed by greedily assigning to each missing pixel the most likely color among those patches from the database that are closest to the current solution. The joint configuration of these assignments minimizes a predefined energy functional based on the (dis-)agreement of overlapping patches. The method of Shen et al. [2006] tries to retain the advantages of global approaches while reducing computational complexity. They track every pixel in the occluded object through the video such that during the energy minimization stage the search space for each pixel is reduced.

Some methods take inspiration from the shift map method for images [Pritch et al. 2007]. This method produces a vector field or shift-map that assigns an offset to every pixel such that the resulting field minimizes a task-dependent energy functional. For image completion, the offsets point to pixels outside the hole from which colors should be copied. An MRF is then optimized that minimizes the discrepancy between the (original) neighborhoods of adjacent pixels. Hu et al. [2010] apply the concept of shift-maps to the video domain. For video retargeting, they obtain a coherent mapping from the original video to a domain of different resolution by maximizing temporal consistency in addition to spatial consistency. Offsets along the time axis are not allowed nor required during the retargeting. However, completing video parts by simply extending shift maps to allow temporal offsets can produce spurious artifacts since, in general, spatial and temporal dimensions have different characteristics. The latter is explicitly considered in the approach by Granados et al. [2012].

The second main class of approaches formulates inpainting as an optimization problem to be solved locally (in space-time) [Jia et al. 2005]. Patwardhan et al. [2007,2009] propose a local approach. They assign a priority to every hole pixel based on the local amount of undamaged pixels and on the presence of structure at the current hole boundary. While this algorithm enables fast completion, in general the results are not guaranteed to be globally coherent.

In Shih et al. [2009], this work is extended to handle general camera motions and remove temporal discontinuities. A more indirect approach is motion transfer which derives a motion field for the hole by gradually propagating motion vectors [Matsushita et al. 2006], or by using motion patch similarities [Shiratori et al. 2006]. The motion field is used to propagate pixel values from outside the hole into the missing region. These approaches allow completion only over a relatively small number of frames.

As mentioned earlier, applying PDE-based per frame inpainting to fill holes of short duration has



Inpainting Method – Granados et al. EG 2012

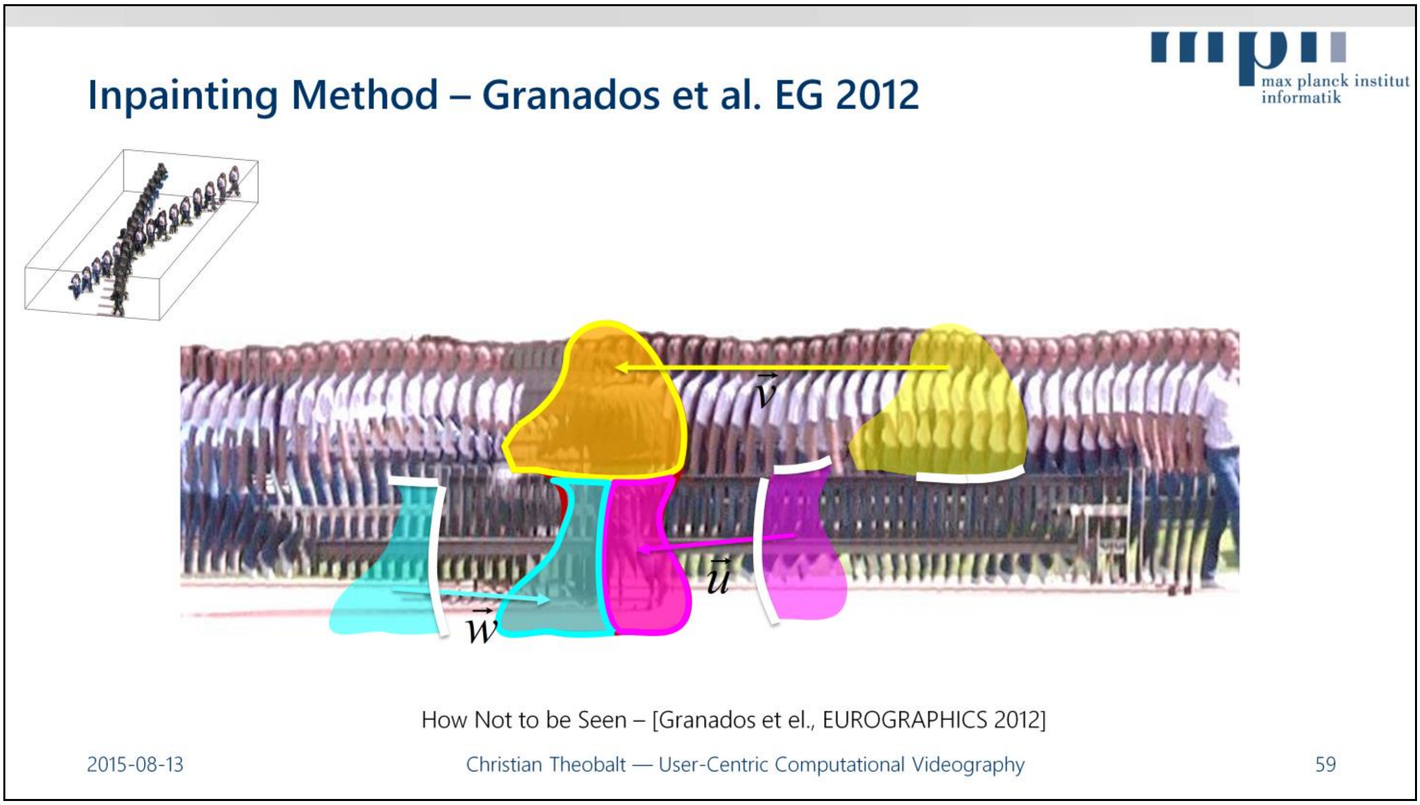
Correspondence-maps [Demanet et al. 2003]
Shift-maps [Pritch et al. 2009]

How Not to be Seen – [Granados et el., EUROGRAPHICS 2012]

In the following, we explain the algorithmic core of the video inpainting approach by Granados et al. [EG 2012]. In order to do this, let us take a front view of the video volume. the person we want to remove is encircled in red here.

We assume that there is a high degree of redundancy in videos, which means that every missing pixel can be filled using some other pixel somewhere else in the video. Pixels in the missing region that are close by are also likely to be replaced with source pixels outside the hole that are also close.

Instead of computing absolute pixel locations, we compute offsets, so that adjacent pixels can have the same offset. This offset field is known as a “correspondence map” or a “shift map”, and has previously been used for image inpainting [Pritch et al. 2009].



Inpainting Method – Granados et al. EG 2012

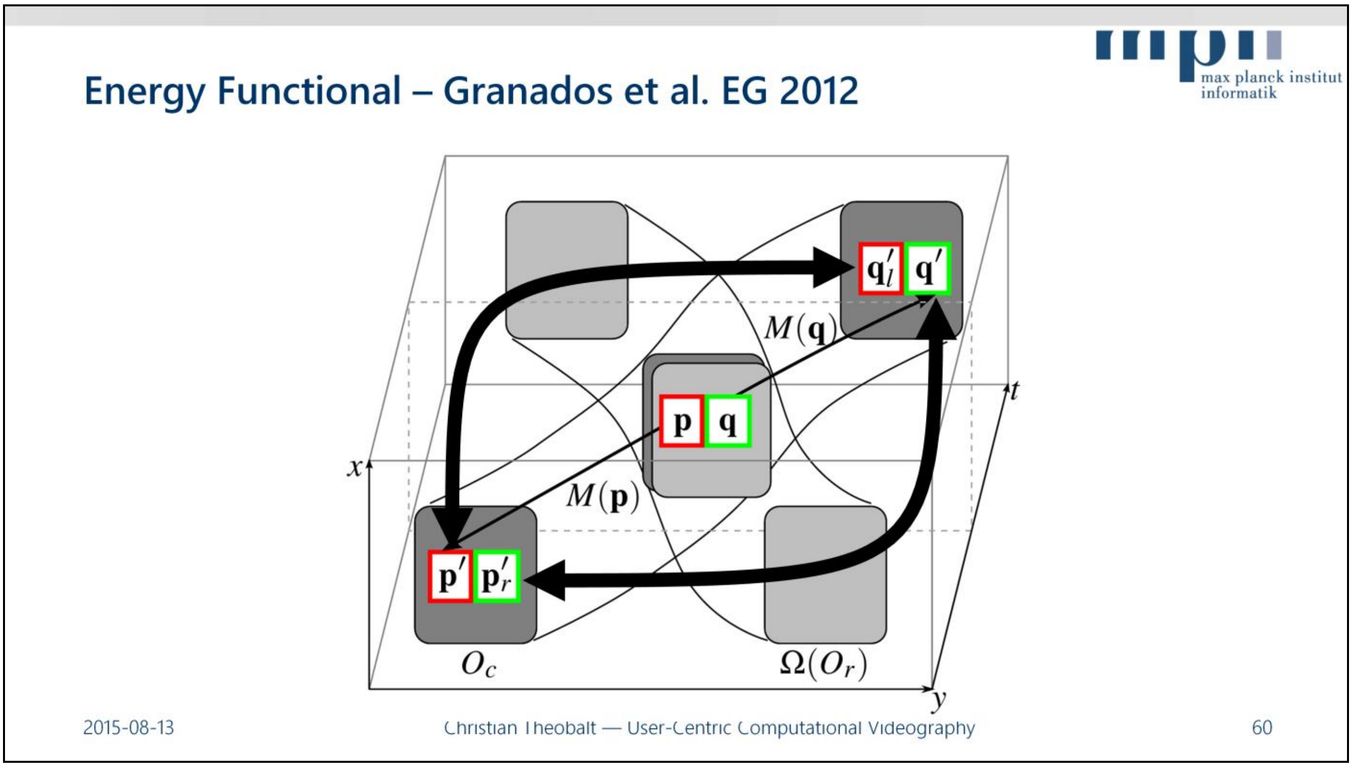
max planck institut informatik

How Not to be Seen – [Granados et el., EUROGRAPHICS 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 59

However, finding a single large region to fill each hole restricts the quality of the final inpainting, as the filled video content needs to be plausible in each frame but also coherent over time. A better approach is to fill the hole with smaller pieces of the video, like in a 3D jigsaw puzzle, where we cut pieces from everywhere in the video, no matter what size or shape they have. The only restriction is that the boundary of each piece has to match those of all the adjacent pieces.

Formulating the problem in such a way is a compromise. Ideally one wants to “shift” as large as possible regions of the original video into the hole, as they are more likely to be spatially coherent. However, too large regions are too inflexible and do not allow to reproduce the dynamics of the inpainted region to be reproduced in a spatially and temporally coherent way.



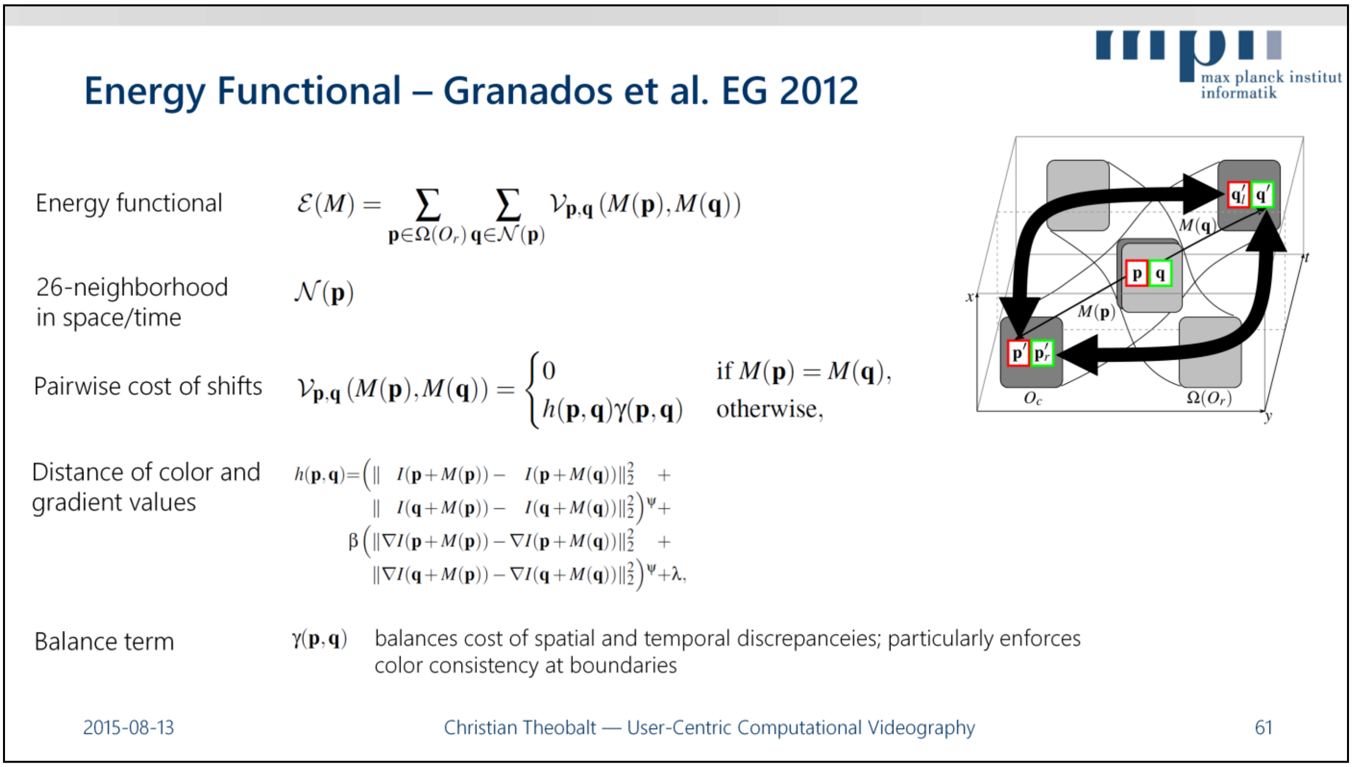
Energy Functional – Granados et al. EG 2012

2015-08-13 Christian Theobalt — User-Centric Computational Videography 60

The criteria and ideas which were explained graphically on the two previous slides are encoded mathematically as an energy functional that is minimized. The solution is a shift map displacement field **M** which, for each pixel in the hole in the video volume, points to the video volume coordinate (in the unoccluded region) from where its content is to be taken.

From a high level, the energy function encodes consistency in a very simple way. Whenever two pixels **p**, **q** have different offsets, it is encouraged that the corresponding sources have similar local appearance. This is done by penalizing the color and gradient differences between corresponding pixels on the two sources. We solve this discrete energy minimization problem (discrete offsets) of an MRF (Markov random field) using graph cuts, and by design, the optimization guarantees solutions that are close to the optimal solution.

See the next slide and the original paper of Granados et al. for the specifics of the energy, as well as the explanation of the latter statement about optimality.



Energy Functional – Granados et al. EG 2012

Energy functional
$$\mathcal{E}(M) = \sum_{\mathbf{p}\in\Omega(O_r)} \sum_{\mathbf{q}\in\mathcal{N}(\mathbf{p})} \mathcal{V}_{\mathbf{p},\mathbf{q}}\left(M(\mathbf{p}), M(\mathbf{q})\right)$$

26-neighborhood in space/time
$$\mathcal{N}(\mathbf{p})$$

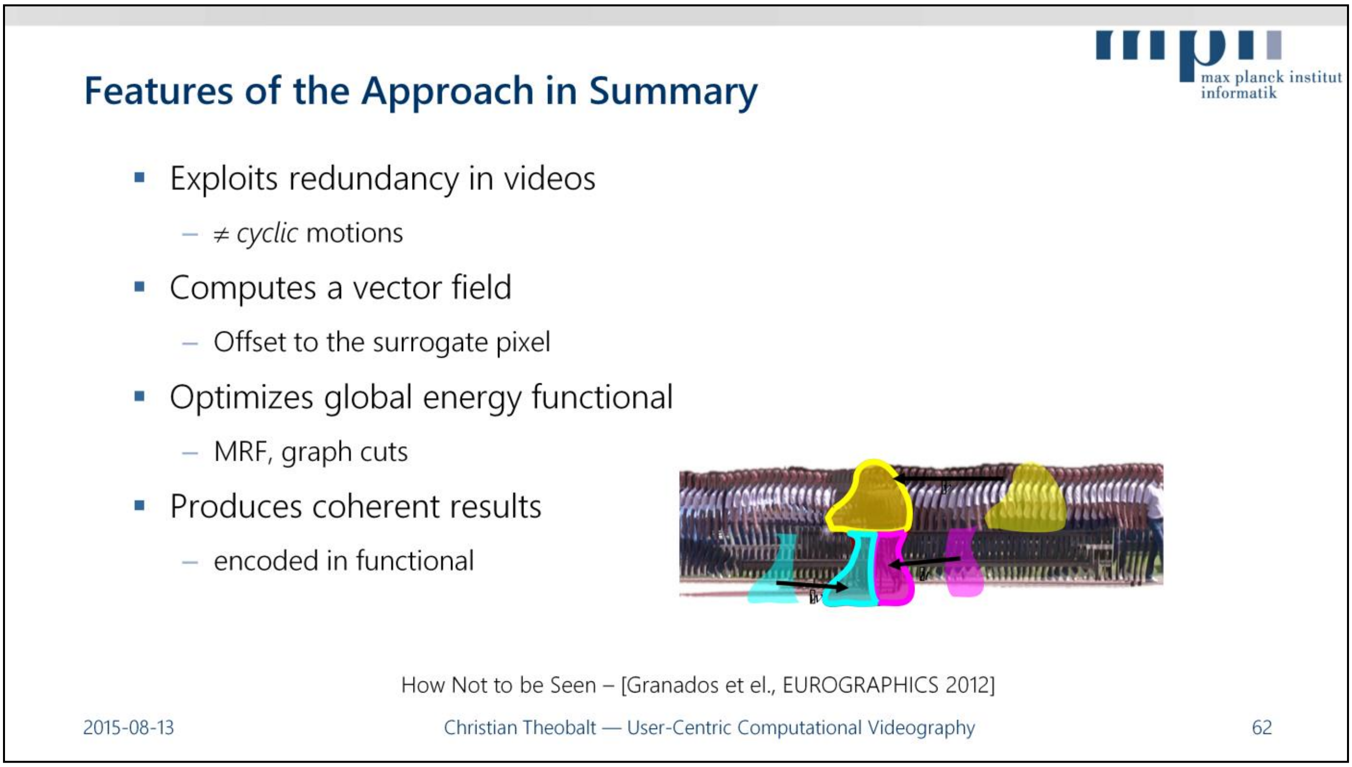
Pairwise cost of shifts
$$\mathcal{V}_{\mathbf{p},\mathbf{q}}\left(M(\mathbf{p}), M(\mathbf{q})\right) = \begin{cases} 0 & \text{if } M(\mathbf{p}) = M(\mathbf{q}), \\ h(\mathbf{p},\mathbf{q})\gamma(\mathbf{p},\mathbf{q}) & \text{otherwise,} \end{cases}$$

Distance of color and gradient values
$$\begin{aligned} h(\mathbf{p},\mathbf{q}) = \Big(& \| \; I(\mathbf{p}+M(\mathbf{p})) - \; I(\mathbf{p}+M(\mathbf{q}))\|_2^2 \; + \\ & \| \; I(\mathbf{q}+M(\mathbf{p})) - \; I(\mathbf{q}+M(\mathbf{q}))\|_2^2 \Big)^{\Psi} + \\ \beta \Big(& \|\nabla I(\mathbf{p}+M(\mathbf{p})) - \nabla I(\mathbf{p}+M(\mathbf{q}))\|_2^2 \; + \\ & \|\nabla I(\mathbf{q}+M(\mathbf{p})) - \nabla I(\mathbf{q}+M(\mathbf{q}))\|_2^2 \Big)^{\Psi} + \lambda, \end{aligned}$$

Balance term $\gamma(\mathbf{p},\mathbf{q})$ balances cost of spatial and temporal discrepanceies; particularly enforces color consistency at boundaries

2015-08-13 Christian Theobalt — User-Centric Computational Videography 61

This slide shows the energy function that is optimized in a bit more detail. A comprehensive explanation of each term, and the algorithmic validation of each term’s relevance is shown in the original paper manuscript.



Features of the Approach in Summary

max planck institut informatik

- Exploits redundancy in videos
  - ≠ *cyclic* motions
- Computes a vector field
  - Offset to the surrogate pixel
- Optimizes global energy functional
  - MRF, graph cuts
- Produces coherent results
  - encoded in functional

How Not to be Seen – [Granados et el., EUROGRAPHICS 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 62

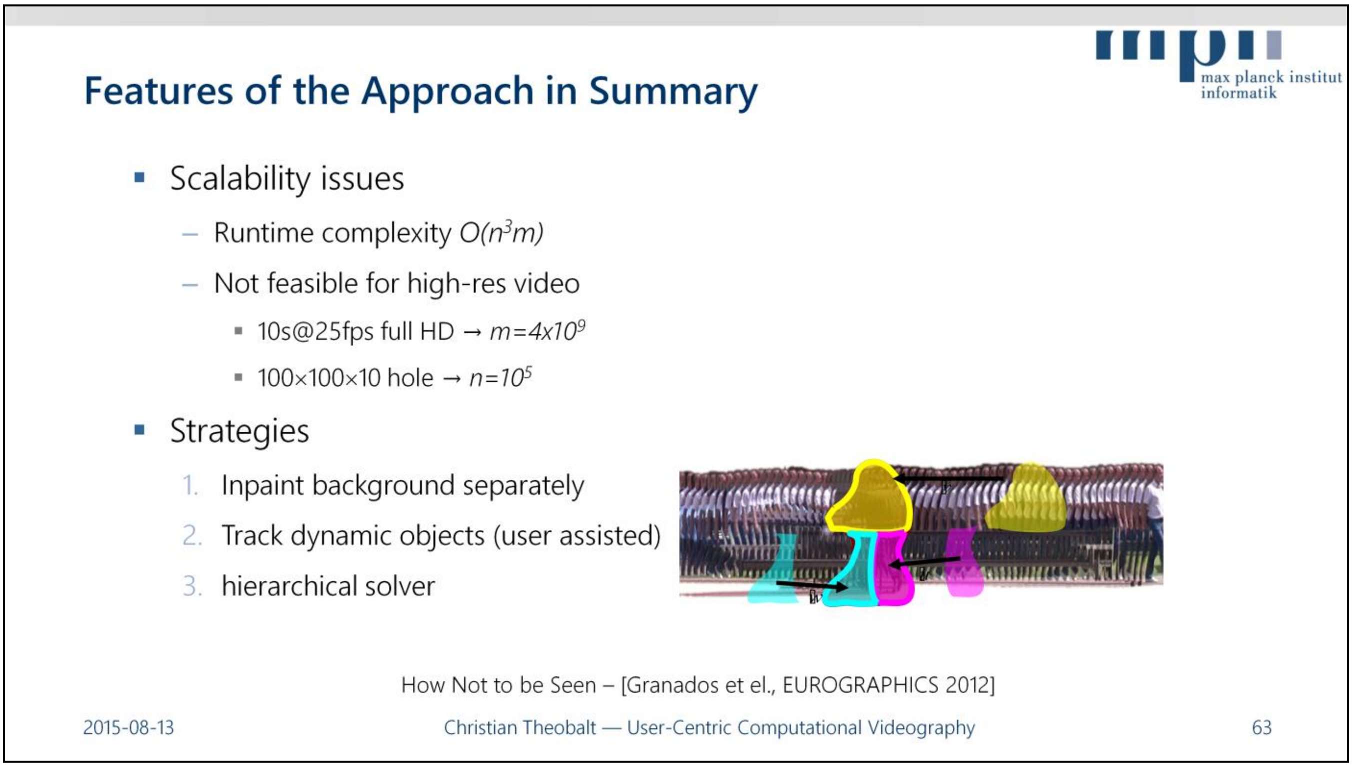
This slide summarizes the main features of the approach by Granados et al.

-It exploits redundancy in videos, and this does not mean assuming cyclic motions.

-It compute a vector field that represent offsets from missing pixels to visible regions.

-It defines the vector field as the minimum of an energy functional, and optimize it using graph cuts.

-It produce coherent results, because coherence is encoded in the functional, as shown in the previous slide



Features of the Approach in Summary

max planck institut informatik

- Scalability issues
  - Runtime complexity $O(n^3m)$
  - Not feasible for high-res video
    - 10s@25fps full HD → $m=4x10^9$
    - 100×100×10 hole → $n=10^5$
- Strategies
  1. Inpaint background separately
  2. Track dynamic objects (user assisted)
  3. hierarchical solver

How Not to be Seen – [Granados et el., EUROGRAPHICS 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 63

Scalability is an important issue that is explicitly addressed. The complexity of Granados' algorithm is O(n^3*m), n is the number of missing pixels, and m is the number of possible offsets. In practice, if one would run this method on high resolution video for a large hole, there would be way too many possible offsets for the computation to be finished. For instance, for 10 seconds of Full HD video at 25fps we have around 4.1x10^9 possible offsets, and say that the hole is just 100x100 pixels for 10 frames, this is 100.000 missing pixels.

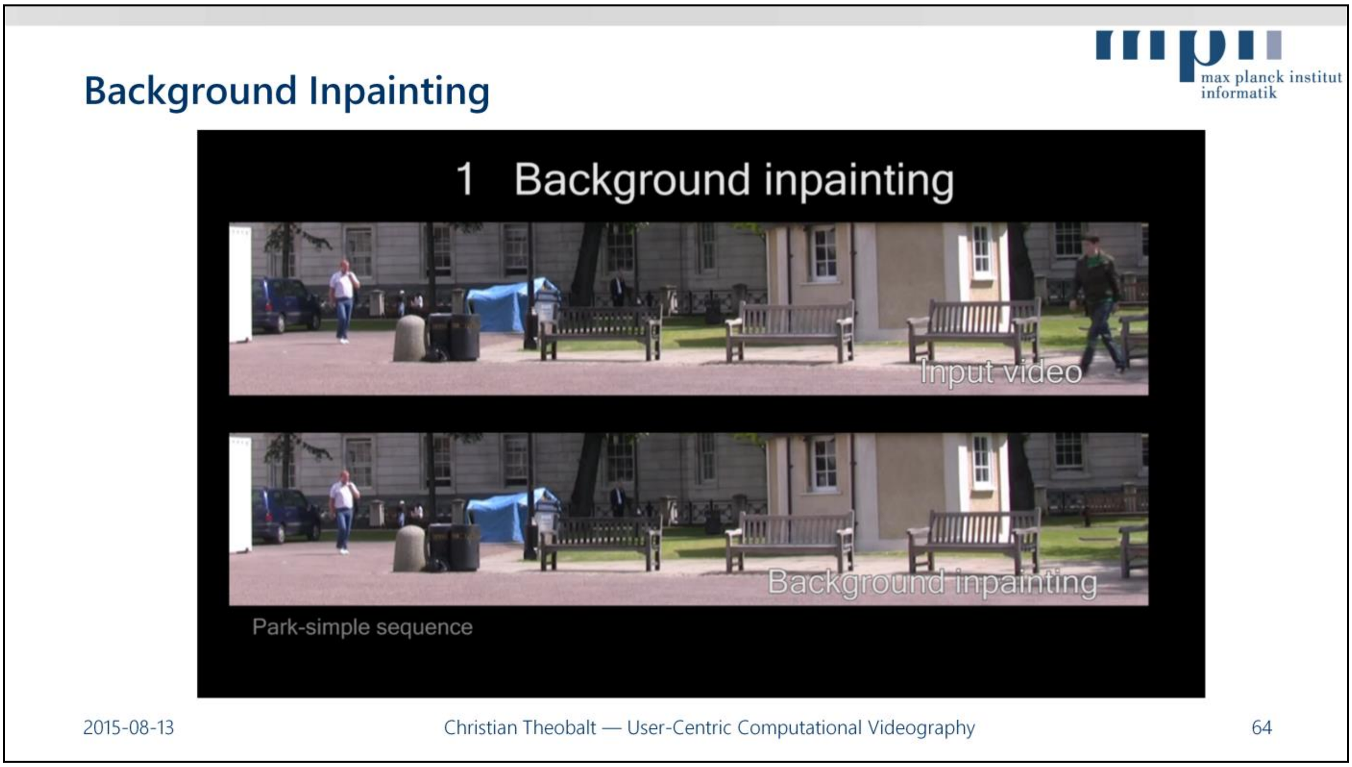
In order to run the method on HD video, one needs to reduce both the size of the hole and the number of offsets as much as possible. To this end, the algorithm relies on three strategies:

1. The static background is inpainted separately, so one can locally concentrate on the more difficult dynamic occlusions.

2. For those dynamic occlusions, the algorithm relies on help from the user to track the relevant region (see next slide).

3. The algorithm performs a hierarchical optimization on a pyramid representation: it computes a solution considering all offsets in coarser resolution, and then refines the solution with only local adjustments in finer resolutions.

Background Inpainting

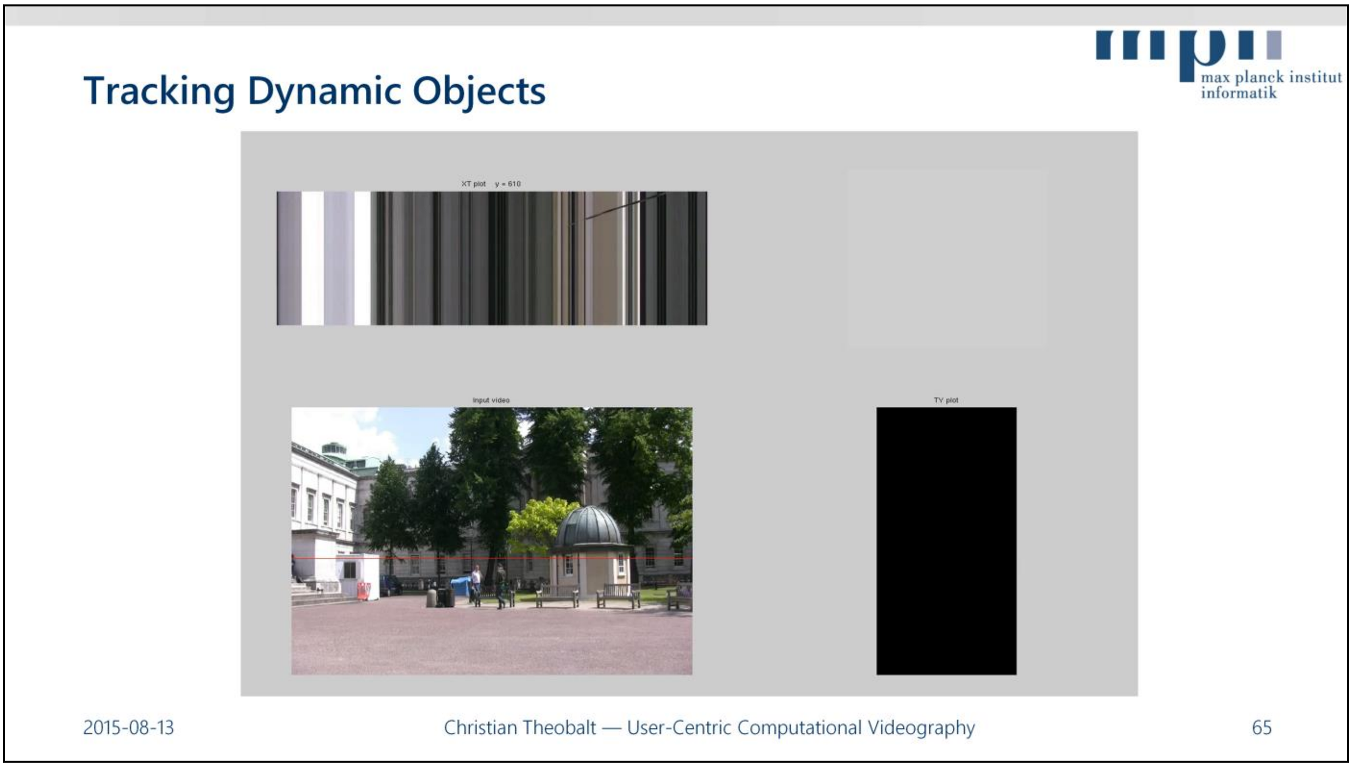


2015-08-13 Christian Theobalt — User-Centric Computational Videography 64

So first we inpaint the part of the scene that is background. Here we can see that since the camera is static, so we only need to consider offsets along the temporal axis, because what we need to copy is located in the same point in space, but on a different frame. This reduces the possible offsets to just twice the number of frames, so the optimization is feasible.

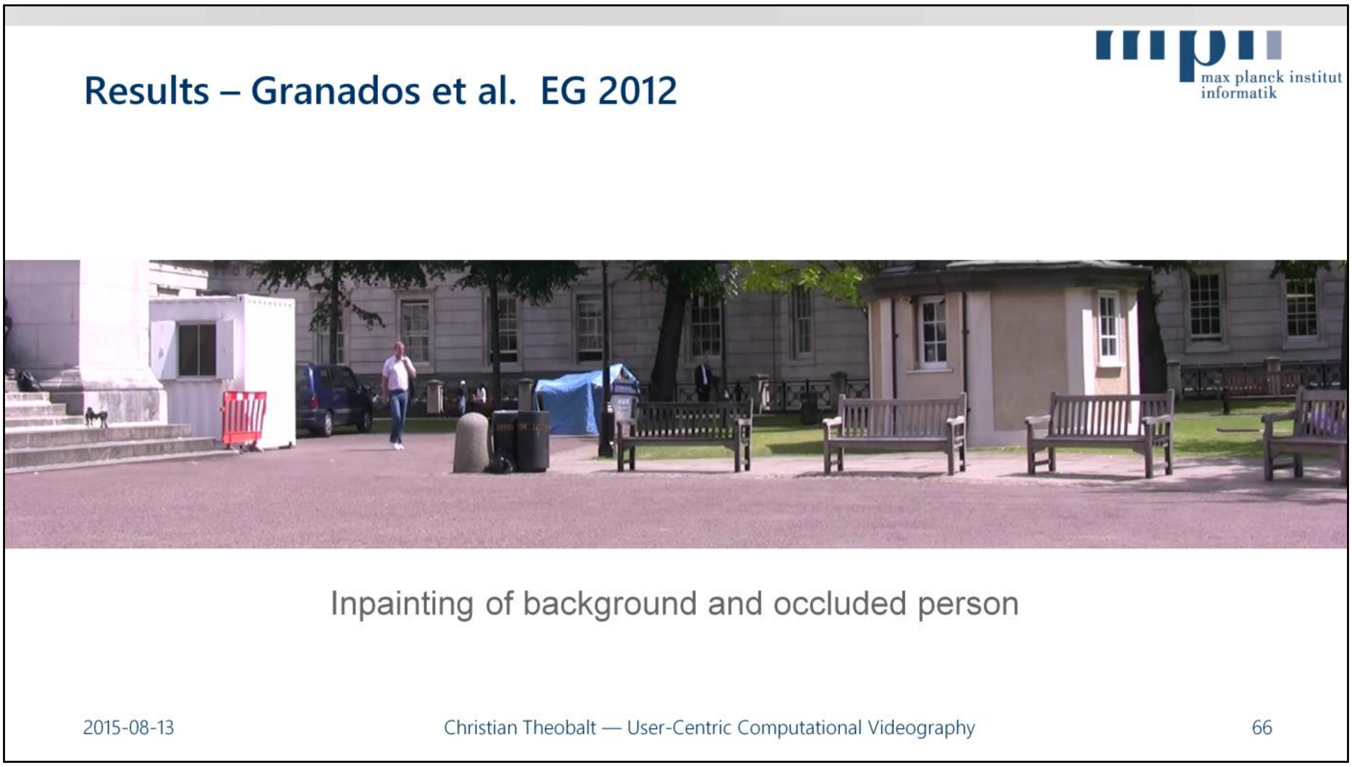
Here we can see that the background estimation is good, but the part where they cross is no yet correct, because the offsets to the right locations were not available to the optimizer at this stage. For making this dynamic part feasible without having to consider all possible locations of the video volume as possible sources of a shift, we use some help from the user.

Video link: http://gvv.mpi-inf.mpg.de/projects/vidinp/



Tracking Dynamic Objects

Video link: http://gvv.mpi-inf.mpg.de/projects/vidinp/

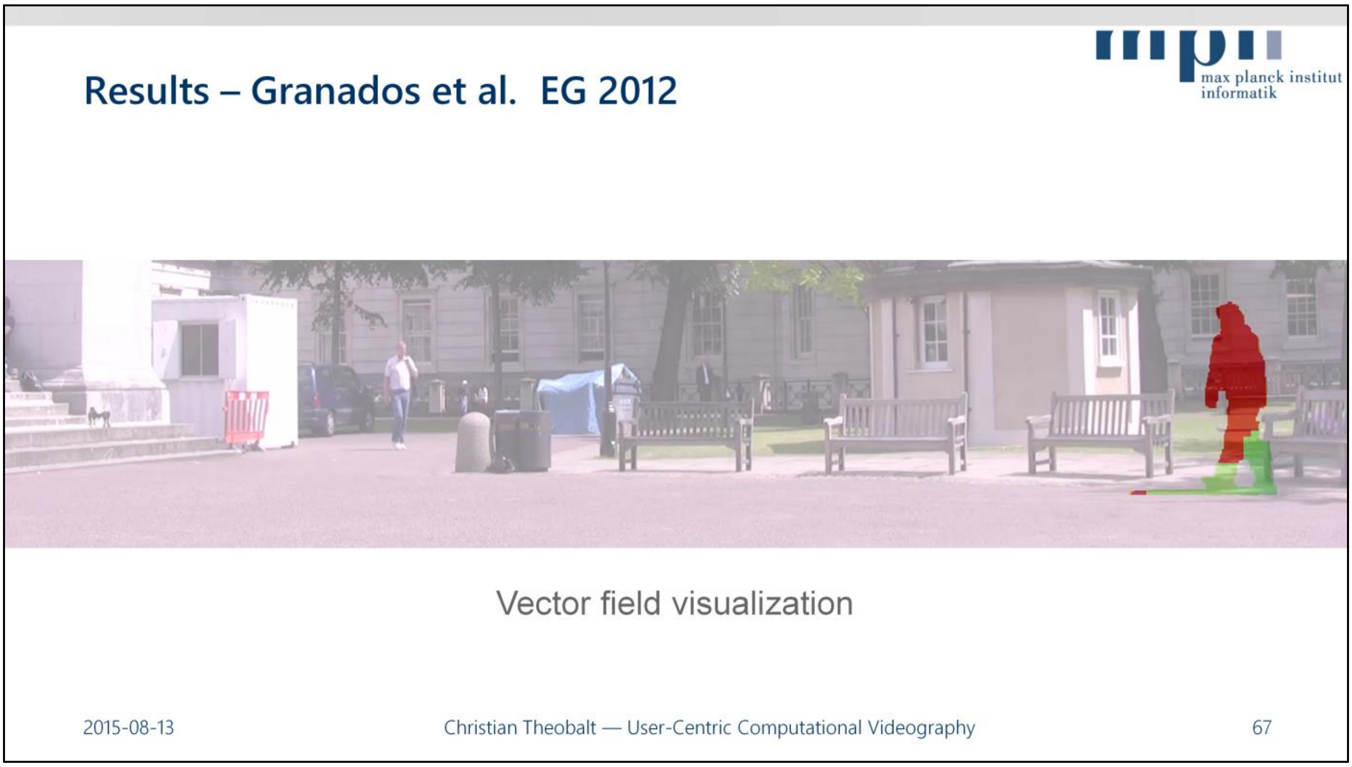


Results – Granados et al. EG 2012

Inpainting of background and occluded person

This is the result for the previously shown sequence with the background and dynamic inpainting together. If you notice a fly crossing right after the occlusion, that's not an artifact, that's just nature. The runtime for this sequence is 40h (on a Xeon X5560 CPU). This is a long runtime but we have to consider that this might be the closest we can get to an optimal solution, if we follow the non-parametric framework with less restrictions on scene type, and this is part of what we wanted to test, to see how far this framework can be pushed. There are ample possibilities to further speed-up the computation of the solution. Also, since the current state of the art to fulfill such a task in practical movie productions is to perform such inpainting tasks manually on each frame, even this long runtime is competitive.

Video link: http://gvv.mpi-inf.mpg.de/projects/vidinp/



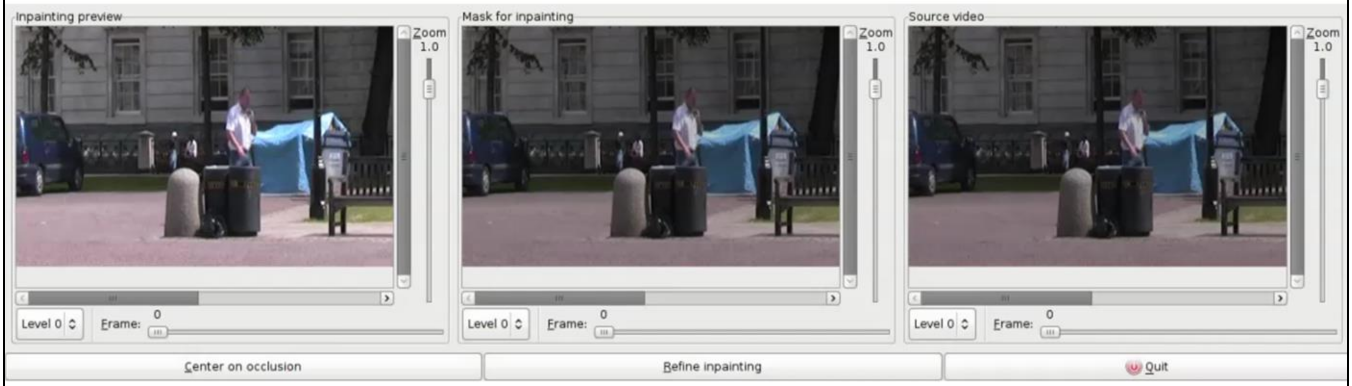
Results – Granados et al. EG 2012

Vector field visualization

This is a visualization of the vector field (shift map field) that is computed. It shows just the index of the source frame for every missing pixel. So different colors mean different sources.

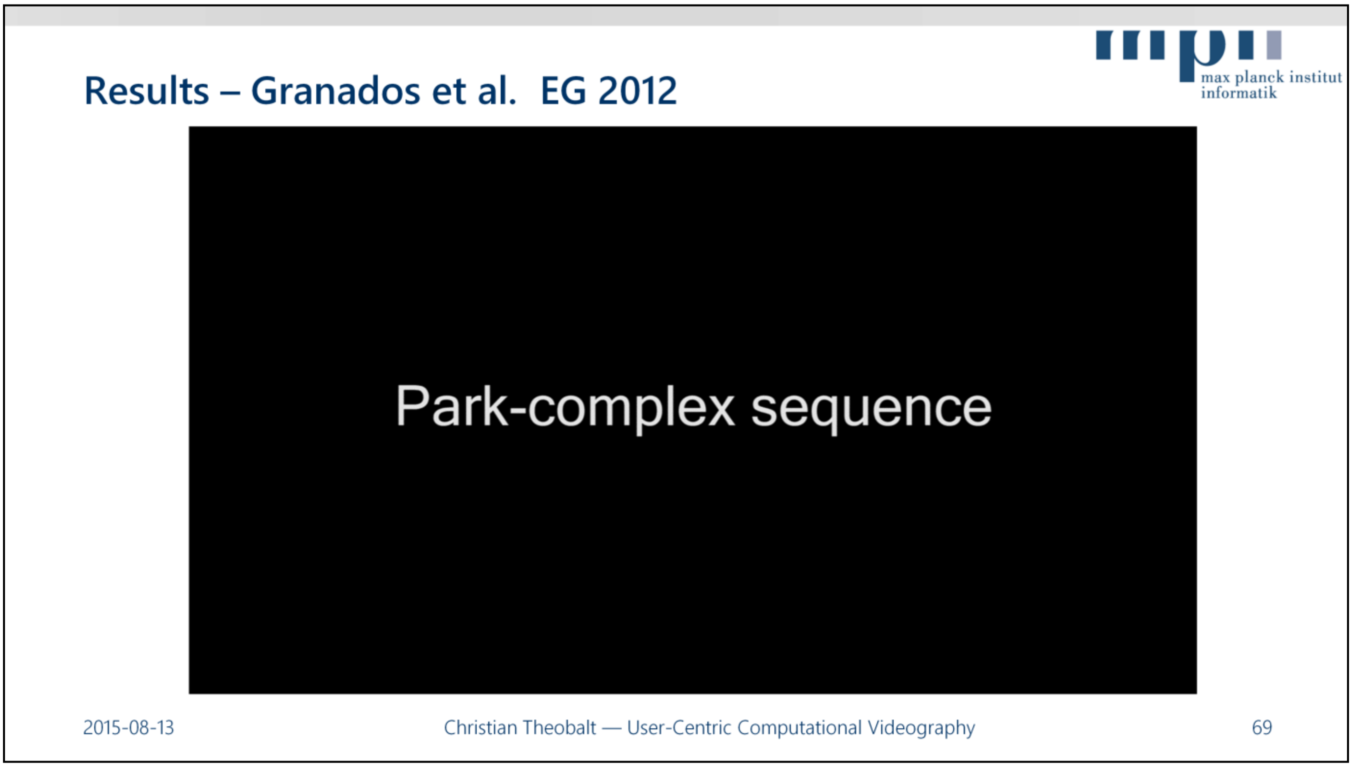
Video link: http://gvv.mpi-inf.mpg.de/projects/vidinp/

User-Guided Refinement



In cases where the algorithm does not produce the result the user wants, like in this toy example, the user can correct small mistakes, interactively: To this end, the user selects the region to repair, and the desired sources, and the algorithm quickly computes an inpainting using only those sources. We think it's important to provide this option to the user, but all the other results were computed without user refinement.

Video link: http://gvv.mpi-inf.mpg.de/projects/vidinp/



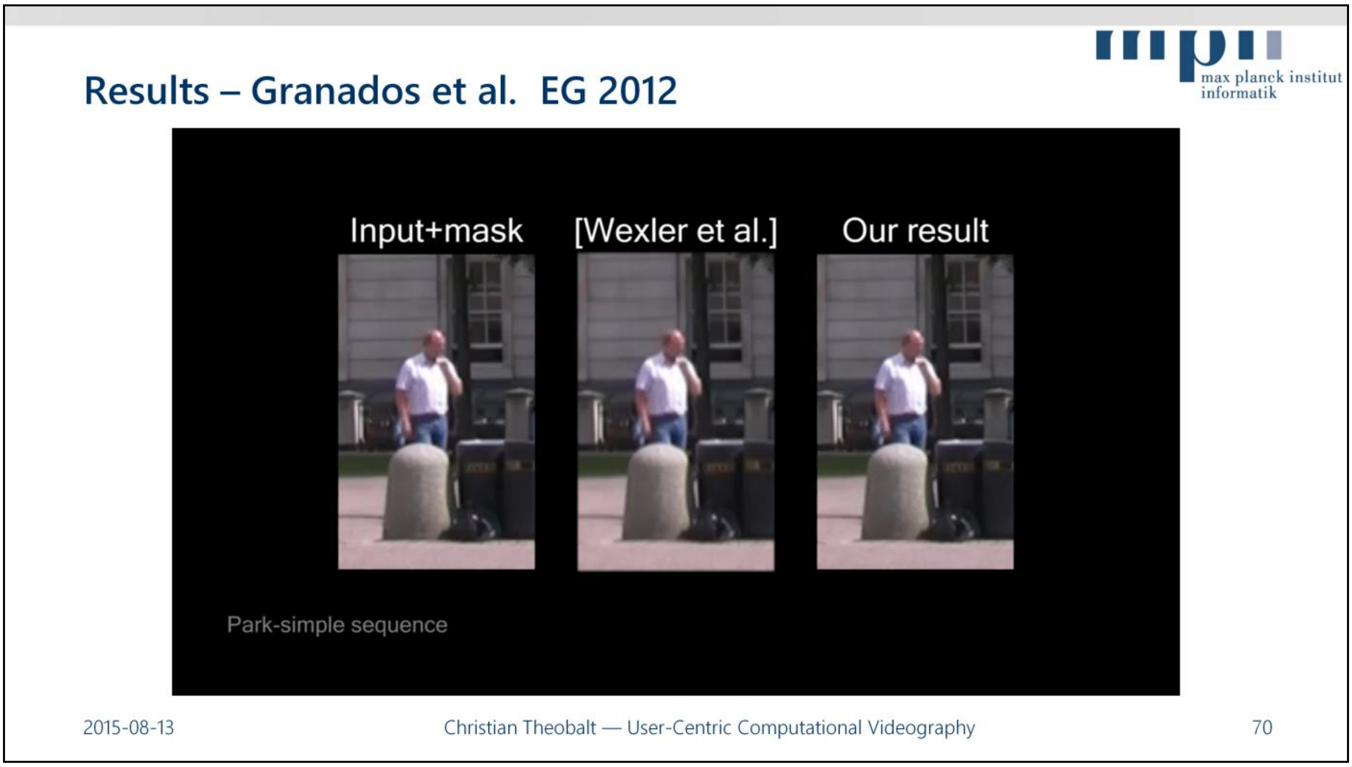
Results – Granados et al. EG 2012

Park-complex sequence

In this sequence, a person that walks in front of several other people with different types of motions was removed. The occlusions of each of these moving persons were inpainted with our method. This kind of diversity in the type of motions and the appearance in inpainted regions wouldn't be possible with a model-based method. The tracking information provided by the user is also shown, along with the resulting inpainting.

The overall result is very plausible, but close inspection will reveal some remaining artifacts. This can happen in to-be-inpainted parts where the missing data is not available elsewhere in the video, for example, when the person in white is walking towards the camera and raising his hand. The method isn't scale invariant, because that would increase the search space quite a lot, so these type of motions are challenging for the method in its current form.

Video link: http://gvv.mpi-inf.mpg.de/projects/vidinp/

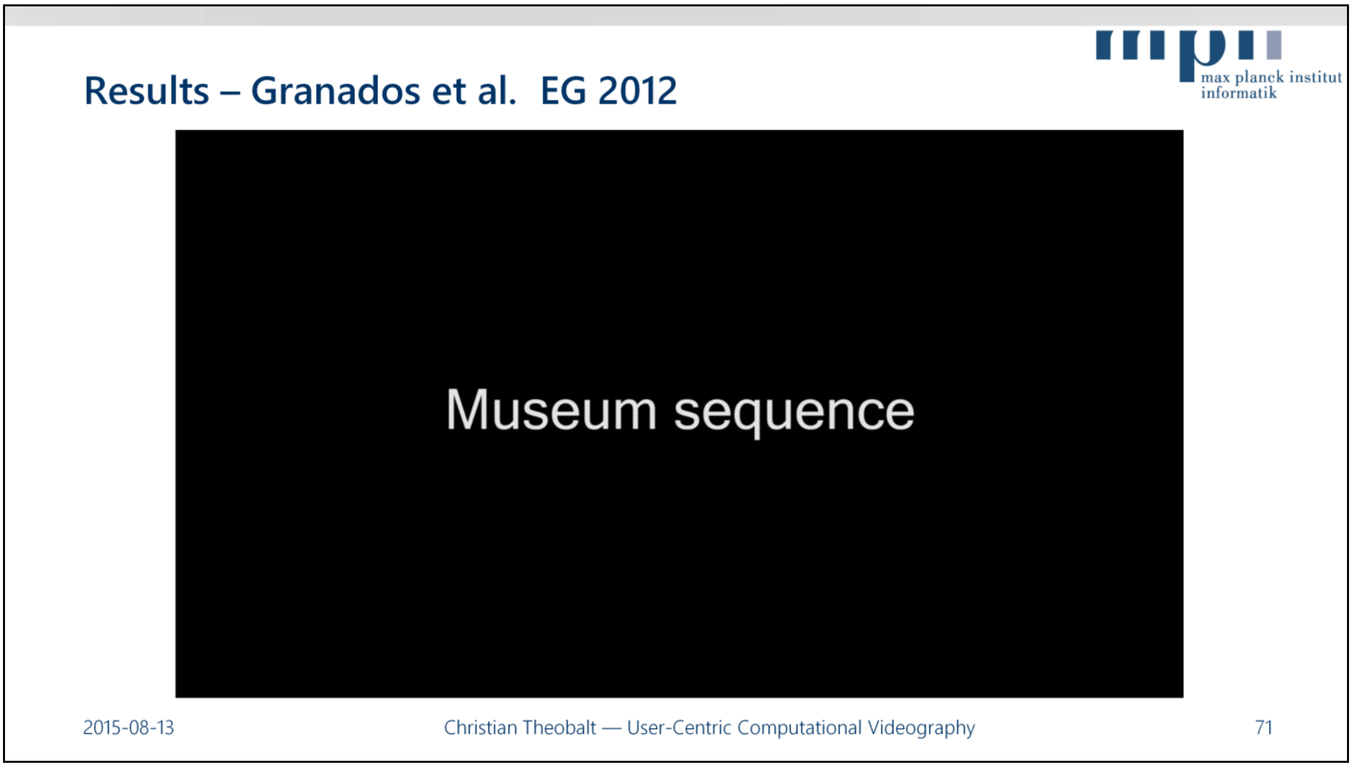


Results – Granados et al. EG 2012

2015-08-13 Christian Theobalt — User-Centric Computational Videography 70

This is a comparison with the method of Wexler et al. As discussed before, they minimize a global energy function, but they use a local optimizer so it can easily get stuck in local minima. This can lead to a wrong inpainting, like the one in the middle, we can't know if is a problem of the energy functional or of the initialization. By design, our optimizer produces solutions within a constant factor of the global minima.

Video link: http://gvv.mpi-inf.mpg.de/projects/vidinp/

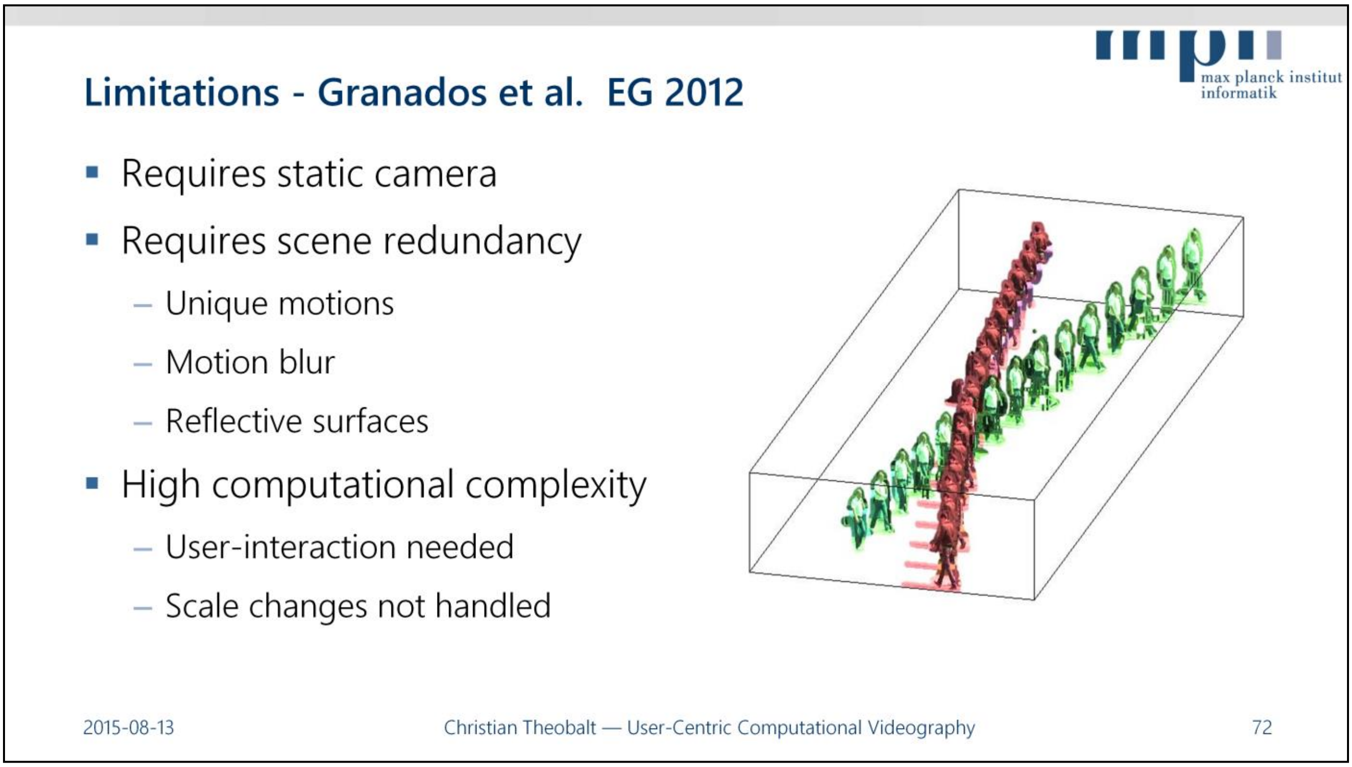


Results – Granados et al. EG 2012

Museum sequence

In this other sequence, a person was removed that is walking over a reflective floor while occluding other people. Overall, the inpainting result is very plausible and several occluded moving persons in the background are correctly inpainted. Some artifacts are visible again in regions where the redundancy assumption is not well satisfied, like when the person on the right with the red bag is walking away from the camera. Artists can use these results directly, or use them as a starting point and optionally remove only some remaining artefacts manually, rather than filling in the entire hole manually as it is often done in practice.

Video link: http://gvv.mpi-inf.mpg.de/projects/vidinp/



Limitations - Granados et al. EG 2012

max planck institut informatik

- Requires static camera
- Requires scene redundancy
  - Unique motions
  - Motion blur
  - Reflective surfaces
- High computational complexity
  - User-interaction needed
  - Scale changes not handled

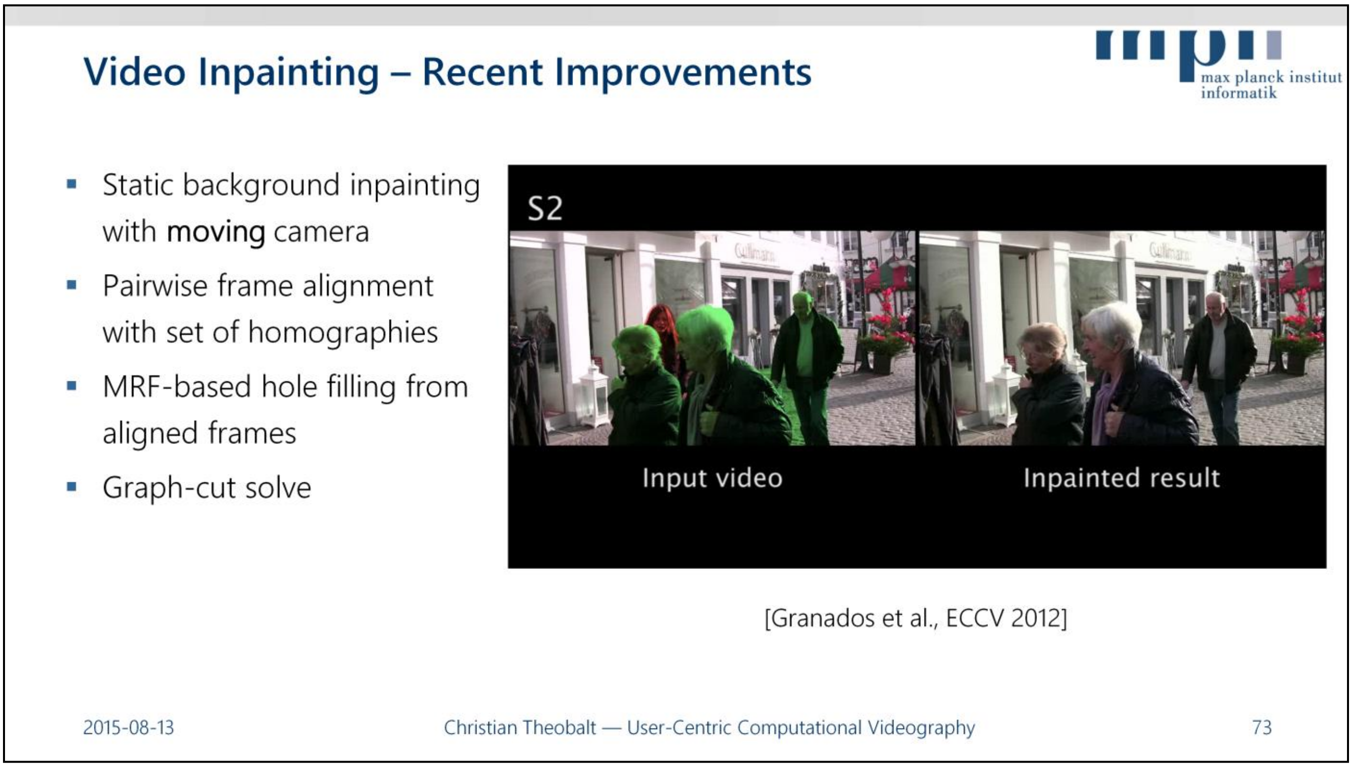
2015-08-13 Christian Theobalt — User-Centric Computational Videography 72

The approach by Granados was a big step ahead over the state-of-the-art in video inpainting. It can be applied for video inpainting on a large range of scenes to which many previous methods were not applicable. It still exhibits several limitations:

-It's designed for static cameras.

-It requires that the video contains enough redundancy so the missing parts can be reconstructed from other bits in the video volume. This is difficult when there are unique motions never seen elsewhere in the video during the occlusion, or when there is motion blur or reflective surfaces where the background and foreground are difficult to separate.

-The computational complexity is high, so we can't handle scale changes, but we can use some user interaction to reduce the runtime.



Video Inpainting – Recent Improvements

- Static background inpainting with **moving** camera
- Pairwise frame alignment with set of homographies
- MRF-based hole filling from aligned frames
- Graph-cut solve

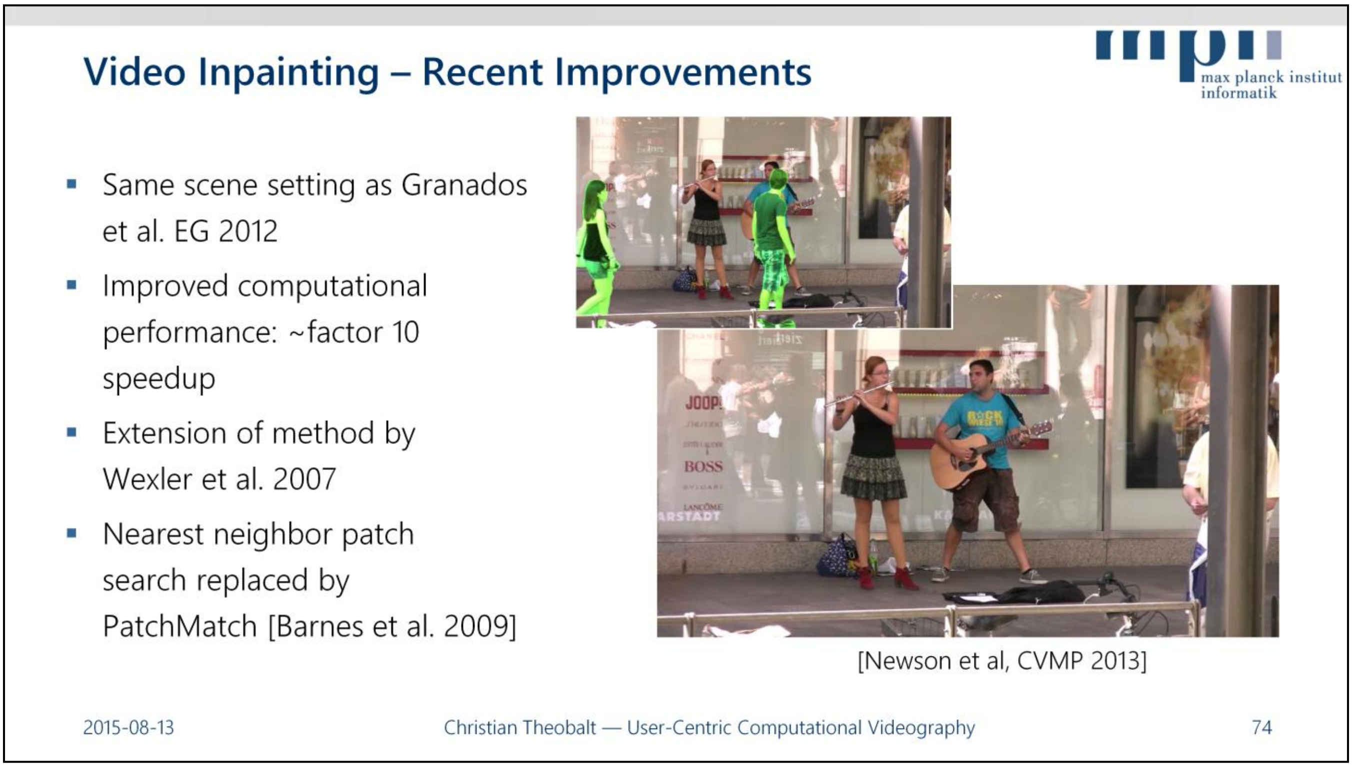
[Granados et al., ECCV 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 73

Some more recently published approaches aim to overcome some of these remaining limitations.

In follow-up work, Granados et al. [ECCV 2012] proposed an approach to inpaint the static background only (so no inpainting of moving background objects) in video filmed with a *moving* camera which would not have been possible with the approach described earlier. To solve this problem, they make a weak assumption about scene geometry, namely that it can be well represented by a set of homographies. This corresponds to assuming that the scene is composed of piecewise planar regions. Based on this homography assumption, all frames are pairwise aligned. After alignment, the hole can be filled from the registered frames by solving an MRF-based energy minimization problem where the best consistent pixels from the aligned frames to be filled into the hole can be found through graph cut optimization. The method produces very plausible inpainting results, only slight temporal artefacts are visible in some of the frames. However, it cannot yet inpaint dynamic background regions in video filmed with a moving camera.

http://gvv.mpi-inf.mpg.de/projects/vidinp/



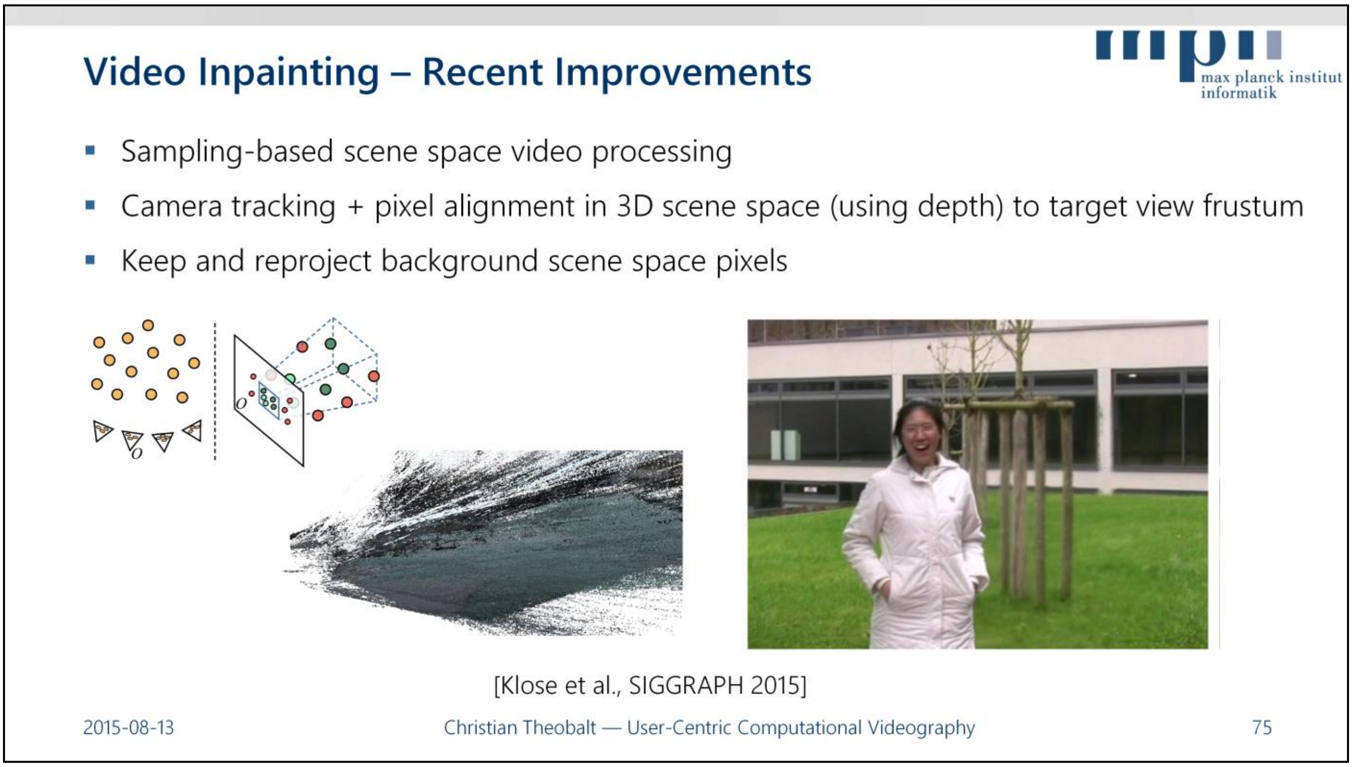
Video Inpainting – Recent Improvements

- Same scene setting as Granados et al. EG 2012
- Improved computational performance: ~factor 10 speedup
- Extension of method by Wexler et al. 2007
- Nearest neighbor patch search replaced by PatchMatch [Barnes et al. 2009]

[Newson et al, CVMP 2013]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 74

One of the main bottlenecks of the approach by Granados et al. [EG 2012] was the quite high computation times of sometimes even several tens of hours for larger holes in high-resolution sequences. Also, the approach relied on some user guidance. To improve computational performance, Newson et al. [2013] proposed an extension of the method by Wexler et al. [2007]. They improve that baseline approach in two ways. First, the time consuming approximate nearest neighbor patch search in the video volume of Wexler et al. is replaced by the PatchMatch randomized patch search algorithm from Barnes et al. [2007]. This leads to a significant speed-up compared to Wexler et al.'s baseline method, but also compared to the approach of Granados et al. Second, Newson et al. improve the patch averaging scheme(s) of Wexler et al. which often caused oversmoothing of the result.



Video Inpainting – Recent Improvements

max planck institut informatik

- Sampling-based scene space video processing
- Camera tracking + pixel alignment in 3D scene space (using depth) to target view frustum
- Keep and reproject background scene space pixels

[Klose et al., SIGGRAPH 2015]

Another approach for video processing based on weak scene model assumptions is shown at SIGGRAPH 2015. The approach is called scene space video processing. The intuition is to align scene samples (so-to-say pixels plus depth) from several frames with a current frame using reconstructed depth and camera calibration information (obtained with off-the-shelf tracking/Structure-from-motion). In a target frame on can inpaint the disoccluded region behind an object by rendering information from the aligned more distant samples of the scene. However, other effects are also possible with this approach.

References Inpainting

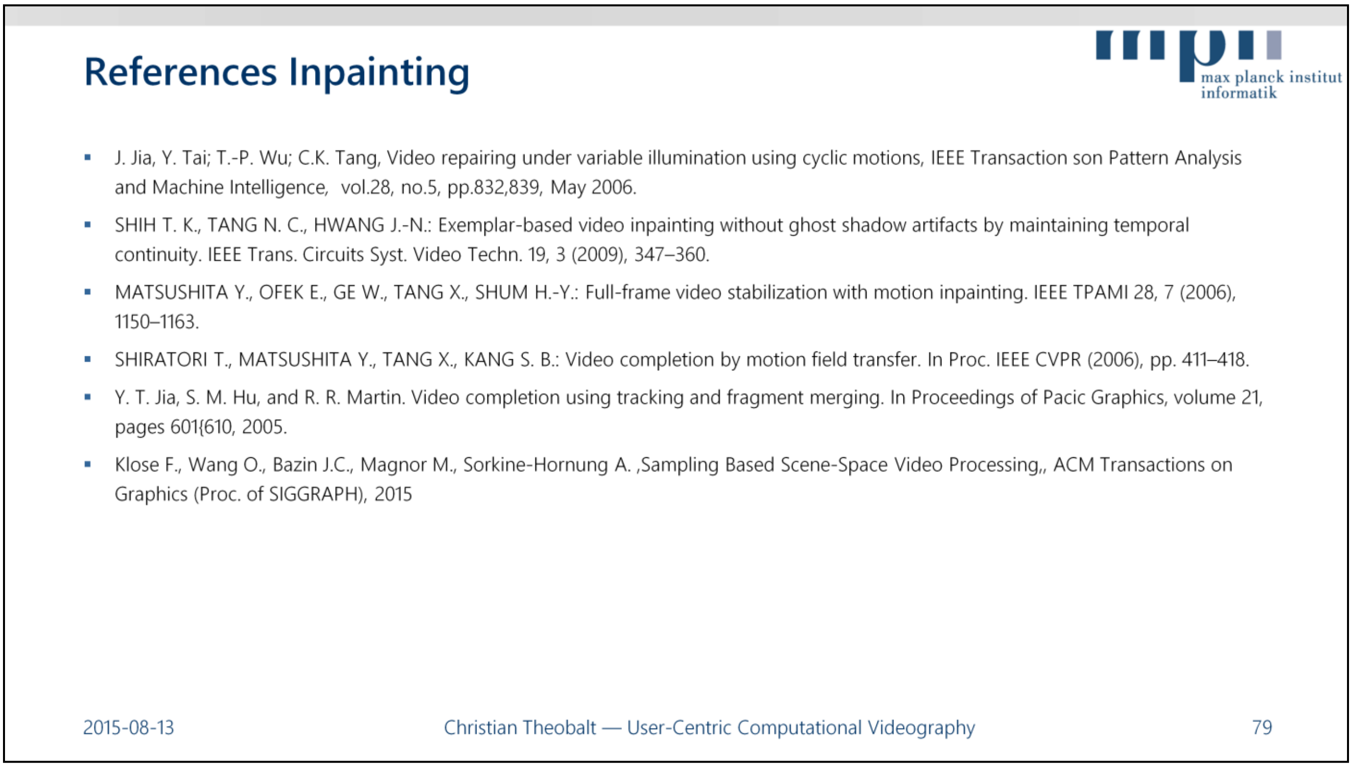
- C. Barnes, E. Shechtman, A. Finkelstein, and DB Goldman. 2009. PatchMatch: a randomized correspondence algorithm for structural image editing. ACM TOG (Proc. SIGGRAPH). 28, 3.
- A. Criminisi, P. Perez, and K. Toyama. 2004. Region filling and object removal by exemplar-based image inpainting. Trans. Img. Proc. 13, 9 (September 2004), 1200-1212.
- A. Efros and W.T. Freeman. Image quilting for texture synthesis and transfer. In Proceedings of ACM Conf. Comp. Graphics (SIGGRAPH), pages 341-346, August 2001.
- D.J. Heeger and J.R. Bergen. Pyramid-based texture analysis/synthesis. In Proceedings of ACM Conf. Comp. Graphics (SIGGRAPH), volume 29, pages 229{233, Los Angeles, CA, 1995.
- H. Igehy and L. Pereira. Image replacement through texture synthesis. In Proceedings of International Conference on Image Processing (ICIP), volume III, pages 186-190, 1997.
- H. Yamauchi, J. Haber, and H.P. Seidel. Image restoration using multiresolution texture synthesis and image inpainting. In Computer Graphics International (CGI 2003).
- M. Bertalmío, G. Sapiro, V. Caselles and C. Ballester., "Image Inpainting", Proceedings of SIGGRAPH 2000, New Orleans, USA, July 2000.
- C.W. Fang and J.J. Lien. Rapid image completion system using multi-resolution patch-based directional and non-directional approaches. IEEE Transactions on Image Processing, 18(11), 2009.
- J. Hays and A. Efros. 2007. Scene completion using millions of photographs. *ACM Trans. Graph.* 26, 3, Article 4 (July 2007)

References Inpainting

- D. Tschumperle and R. Deriche. 2005. Vector-Valued Image Regularization with PDEs: A Common Framework for Different Applications. *IEEE Trans. Pattern Anal. Mach. Intell.* 27, 4 (April 2005), 506-517.
- J. Sun, L. Yuan, J. Jia, and H. Shum. Image completion with structure propagation. In Proceedings of ACM SIGGRAPH 2005.
- M. Oliviera, B. Bowen, R. McKenna, and Y.-S. Chang. Fast digital image inpainting. In Proc. of Intl. Conf. on Visualization, Imaging and Image Processing (VIIP).
- Elad, M., Starck, J.L., Querre, P., Donoho, D.L.: Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA). Appl. Comput. Harmon. Anal. 19(3), 340---358 (2005).
- A. Bugeau and M. Bertalmío, Combining texture synthesis and diffusion for image inpainting**.** VISAPP 2009. International Conference on Computer Vision Theory and Applications, Lisboa (Portugal), 2009.
- M. Bertalmío, A. Bertozzi, G. Sapiro, Navier-Stokes, Fluid-Dynamics and Image and Video Inpainting, IEEE CVPR 2001, Hawaii, USA, December 2001.
- M. Granados, J. Tompkin, K. I. Kim, O. Grau, J. Kautz, C. Theobalt, How Not to Be Seen - Object Removal from Videos of Crowded Scenes in Computer Graphics Forum (Proc. EUROGRAPHICS), 31 (2), 219 - 228 (2012).
- M. Granados, K. I. Kim, J. Tompkin, J. Kautz, C. Theobalt, Background inpainting for videos with dynamic objects and a free-moving camera, in Proc. ECCV, 682 - 695 (2012).

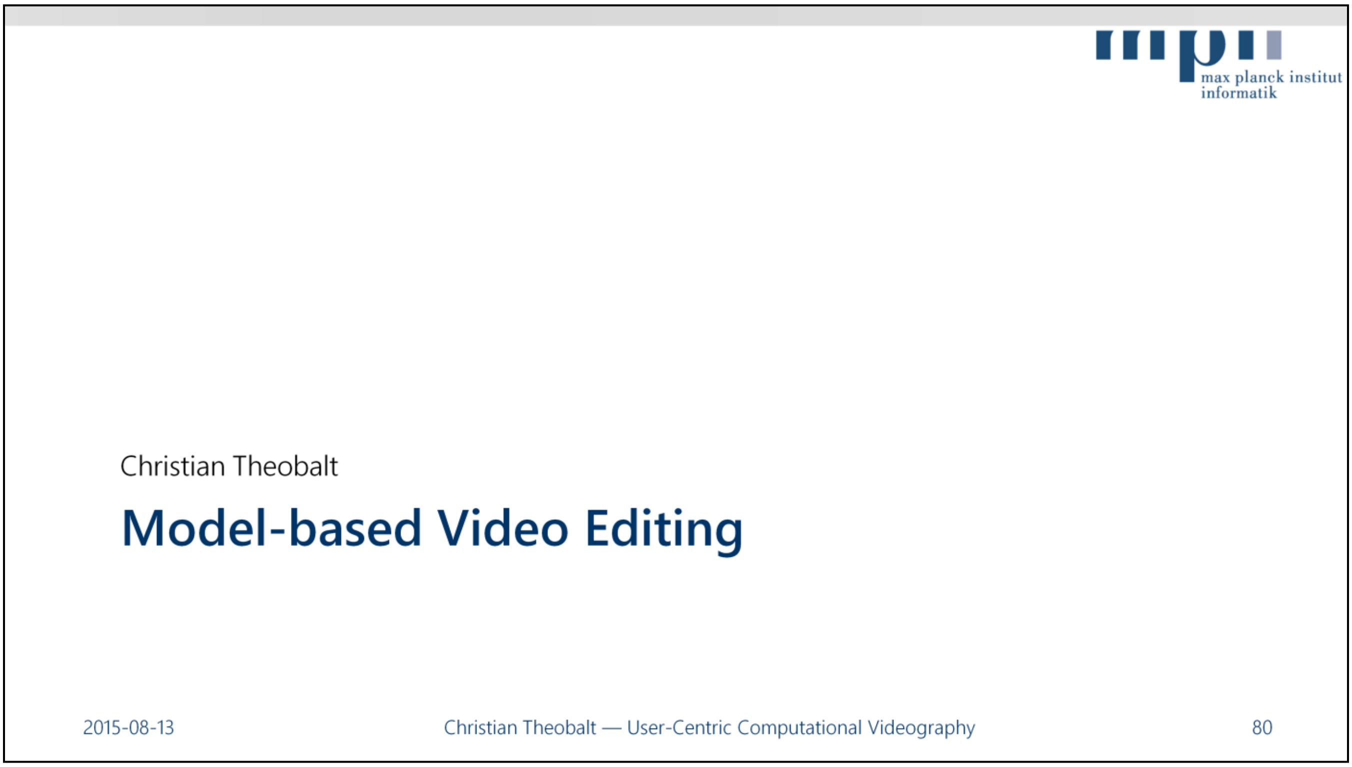
References Inpainting

- J. Jia, Y. Tai; T.-P. Wu; C.K. Tang, Video repairing under variable illumination using cyclic motions, IEEE Transaction son Pattern Analysis and Machine Intelligence, vol.28, no.5, pp.832,839, May 2006.
- A. Newson, A. Almansa, M. Fradet, Y. Gousseau, Perez, Towards fast, generic video inpainting, in Eur. Conf. Visual Media Production (CVMP), 2013.
- VENKATESH M. V., CHEUNG S. S., ZHAO J.: Efficient object-based video inpainting. Pattern Recognition Letters 30, 2 (2009), 168–179.
- LING C.-H., LIN C.-W., SU C.-W., LIAO H.-Y. M., CHEN Y.-S.: Video object inpainting using posture mapping. In Proc. ICIP (2009).
- JIA Y.-T., HU S.-M., MARTIN R. R.: Video completion using tracking and fragment merging. The Visual Computer 21, 8-10 (2005), 601–610.
- WEXLER Y., SHECHTMAN E., IRANI M.: Space-time completion of video. IEEE TPAMI 29, 3 (2007), 463–476.
- SHEN Y., LU F., CAO X., FOROOSH H.: Video completion for perspective camera under constrained motion. In Proc. ICIP (2006), vol. 3, pp. 63–66.
- PRITCH Y., KAV-VENAKI E., PELEG S.: Shift-map image editing. In Proc. ICCV (2009), pp. 151–158.
- HU Y., RAJAN D.: Hybrid shift map for video retargeting. In Proc. IEEE CVPR (2010), IEEE, pp. 577–584.
- PATWARDHAN K. A., SAPIRO G., BERTALMIO M.: Video inpainting of occluding and occluded objects. In Proc. ICIP (2005), pp. 69–72.
- PATWARDHAN K., SAPIRO G., BERTALMIO M.: Video inpainting under constrained camera motion. IEEE TIP 16, 2, (February 2007), 545–553.



References Inpainting

- J. Jia, Y. Tai; T.-P. Wu; C.K. Tang, Video repairing under variable illumination using cyclic motions, IEEE Transaction son Pattern Analysis and Machine Intelligence, vol.28, no.5, pp.832,839, May 2006.
- SHIH T. K., TANG N. C., HWANG J.-N.: Exemplar-based video inpainting without ghost shadow artifacts by maintaining temporal continuity. IEEE Trans. Circuits Syst. Video Techn. 19, 3 (2009), 347–360.
- MATSUSHITA Y., OFEK E., GE W., TANG X., SHUM H.-Y.: Full-frame video stabilization with motion inpainting. IEEE TPAMI 28, 7 (2006), 1150–1163.
- SHIRATORI T., MATSUSHITA Y., TANG X., KANG S. B.: Video completion by motion field transfer. In Proc. IEEE CVPR (2006), pp. 411–418.
- Y. T. Jia, S. M. Hu, and R. R. Martin. Video completion using tracking and fragment merging. In Proceedings of Pacic Graphics, volume 21, pages 601{610, 2005.
- Klose F., Wang O., Bazin J.C., Magnor M., Sorkine-Hornung A. ,Sampling Based Scene-Space Video Processing,, ACM Transactions on Graphics (Proc. of SIGGRAPH), 2015



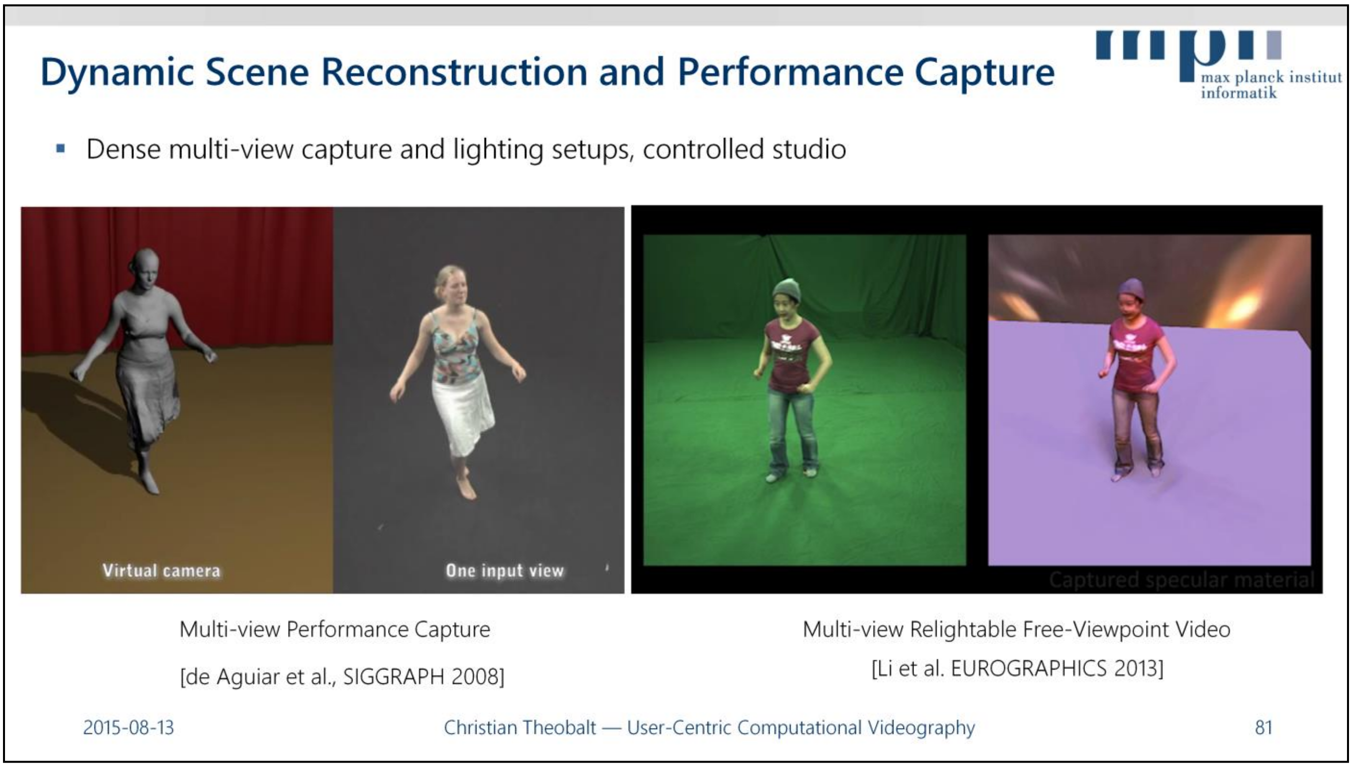
max planck institut informatik

Christian Theobalt

Model-based Video Editing

2015-08-13 Christian Theobalt — User-Centric Computational Videography 80

The approaches reviewed so-far used comparably “lightweight” models of the space-time relationships in videos (e.g. spatial or temporal similarity relations between regions of a video volume) to enable advanced video edits. In the following, we discuss approaches that rely on the availability of much stronger models, for instance more detailed 3D shape, motion and appearance models registered to every frame of video. While this often restricts their use to certain types of scenes, very advanced edits are feasible under these conditions.



Dynamic Scene Reconstruction and Performance Capture

- Dense multi-view capture and lighting setups, controlled studio

Multi-view Performance Capture

[de Aguiar et al., SIGGRAPH 2008]

Multi-view Relightable Free-Viewpoint Video

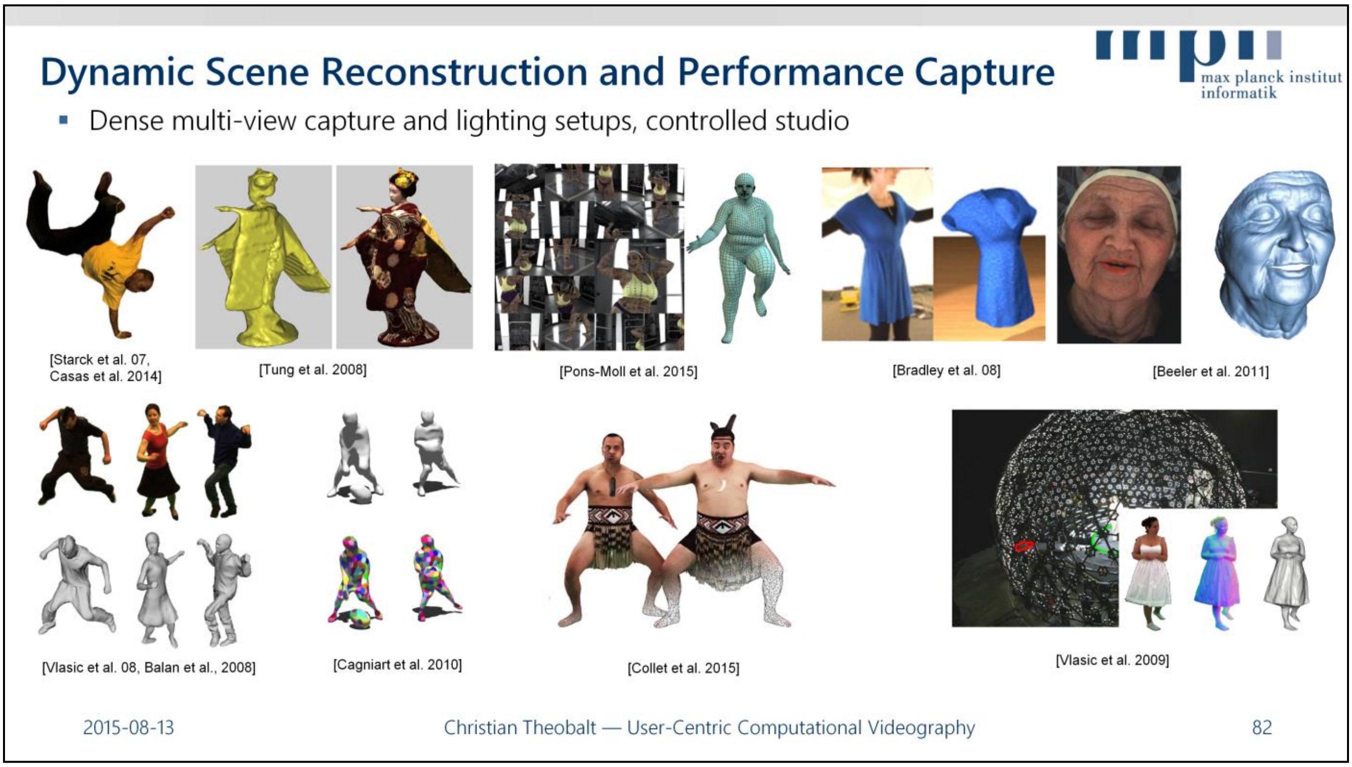
[Li et al. EUROGRAPHICS 2013]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 81

Model-based video editing approaches benefit from converging strands of research in both computer vision and computer graphics. Ever more advanced marker-less algorithms are developed to capture detailed models of shape, motion and geometry of human actors from video recordings. Dynamic scene reconstructions with the best quality are obtained with so-called performance capture approaches that usually require the scene to be capture from multiple camera views in a controlled studio, see above results from de Aguiar et al. [2008] and Li et al. [2013] as an example. Reviewing performance capture methods extensively is beyond the scope of this course, but in the following, we provide a comprehensive list of references in case the reviewer is interested to read up in more detail.

As stated above, many approaches achieve high quality dynamic shape reconstructions by resorting to multi-camera video footage under controlled settings [de Aguiar et al. 2008, Tung et al. 2008, Casas et al. 2014, Pons-Moll et al. 2015, Bradley et al. 08, Beeler et al. 2011, Vlasic et al. 08, Balan et al. 2008, Cagniart et al. 2010, Collet et al. 2015, Vlasic et al. 2009]. For dynamic scene reconstruction, approaches resort to a variety of algorithmic strategies, including reconstruction using templates or statistical shape models [e.g. de Aguiar et al. 2008, Vlasic et al. 2008 etc.], variants of shape-from-silhouette [e.g. Cagniart et al. 2010], variants of multi-view stereo [e.g. Beeler et al. 2011], or variants of photometric stereo [e.g. Vlasic et al. 2009]. In some cases, this makes it necessary that scenes are recorded under a complex controllable lighting setup. See also the following course notes, book chapters and books for more extended reviews on the topic [Theobalt et al. 2007, Theobalt et al. 2008, Theobalt et al. 201, Magnor et al. 2015].

The aforementioned performance capture approaches produce results of high quality, but he strong requirements regarding the acquisition system make them unsuitable for model-based single video editing.



Dynamic Scene Reconstruction and Performance Capture

- Dense multi-view capture and lighting setups, controlled studio

[Starck et al. 07, Casas et al. 2014]

[Tung et al. 2008]

[Pons-Moll et al. 2015]

[Bradley et al. 08]

[Beeler et al. 2011]

[Vlasic et al. 08, Balan et al., 2008]

[Cagniart et al. 2010]

[Collet et al. 2015]

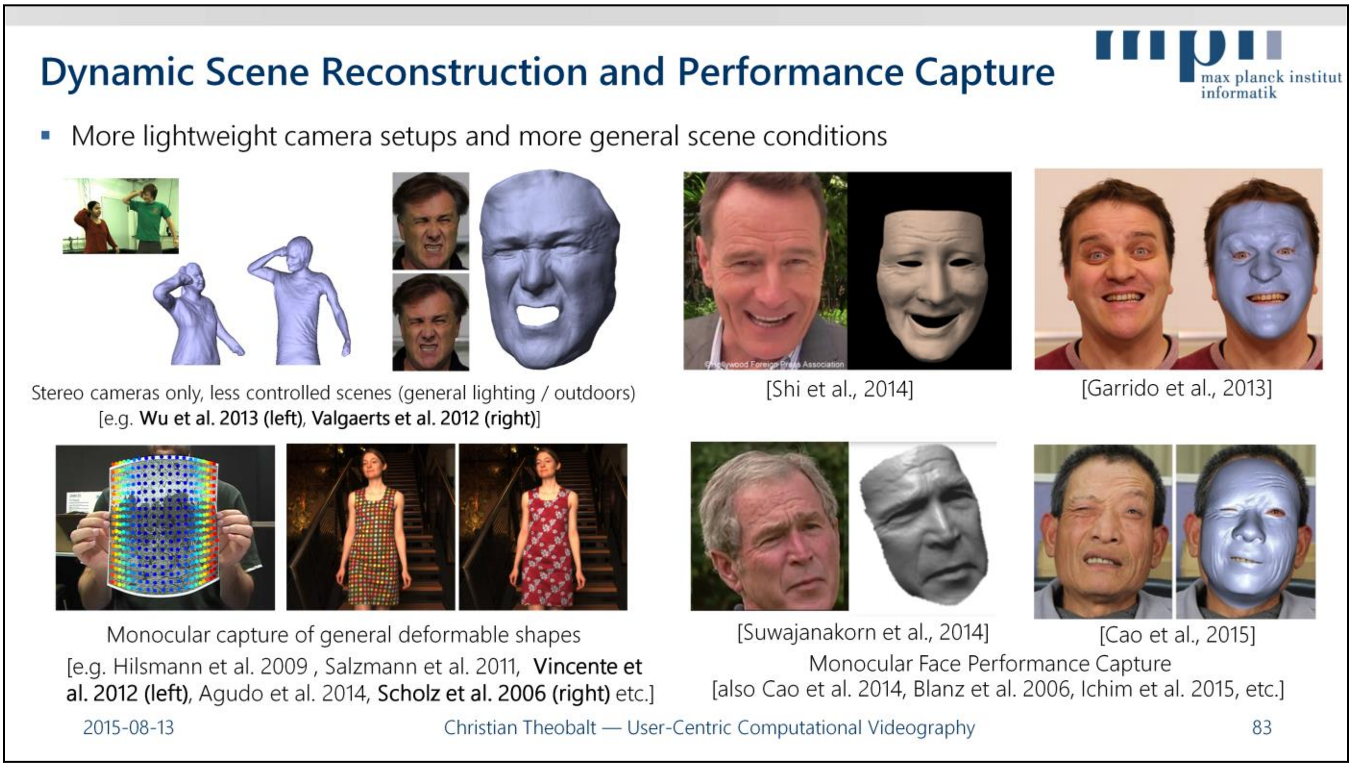
[Vlasic et al. 2009]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 82

Model-based video editing approaches benefit from converging strands of research in both computer vision and computer graphics. Ever more advanced marker-less algorithms are developed to capture detailed models of shape, motion and geometry of human actors from video recordings. Dynamic scene reconstructions with the best quality are obtained with so-called performance capture approaches that usually require the scene to be capture from multiple camera views in a controlled studio, see above results from de Aguiar et al. [2008] and Li et al. [2013] as an example. Reviewing performance capture methods extensively is beyond the scope of this course, but in the following, we provide a comprehensive list of references in case the reviewer is interested to read up in more detail.

As stated above, many approaches achieve high quality dynamic shape reconstructions by resorting to multi-camera video footage under controlled settings [de Aguiar et al. 2008, Tung et al. 2008, Casas et al. 2014, Pons-Moll et al. 2015, Bradley et al. 08, Beeler et al. 2011, Vlasic et al. 08, Balan et al. 2008, Cagniart et al. 2010, Collet et al. 2015, Vlasic et al. 2009]. For dynamic scene reconstruction, approaches resort to a variety of algorithmic strategies, including reconstruction using templates or statistical shape models [e.g. de Aguiar et al. 2008, Vlasic et al. 2008 etc.], variants of shape-from-silhouette [e.g. Cagniart et al. 2010], variants of multi-view stereo [e.g. Beeler et al. 2011], or variants of photometric stereo [e.g. Vlasic et al. 2009]. In some cases, this makes it necessary that scenes are recorded under a complex controllable lighting setup. See also the following course notes, book chapters and books for more extended reviews on the topic [Theobalt et al. 2007, Theobalt et al. 2008, Theobalt et al. 201, Magnor et al. 2015].

The aforementioned performance capture approaches produce results of high quality, but he strong requirements regarding the acquisition system make them unsuitable for model-based single video editing.



Dynamic Scene Reconstruction and Performance Capture

- More lightweight camera setups and more general scene conditions

Stereo cameras only, less controlled scenes (general lighting / outdoors) [e.g. **Wu et al. 2013 (left)**, **Valgaerts et al. 2012 (right)**]

[Shi et al., 2014]

[Garrido et al., 2013]

Monocular capture of general deformable shapes [e.g. Hilsmann et al. 2009 , Salzmann et al. 2011, **Vincente et al. 2012 (left)**, Agudo et al. 2014, **Scholz et al. 2006 (right)** etc.]

[Suwajanakorn et al., 2014]

[Cao et al., 2015]

Monocular Face Performance Capture [also Cao et al. 2014, Blanz et al. 2006, Ichim et al. 2015, etc.]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 83

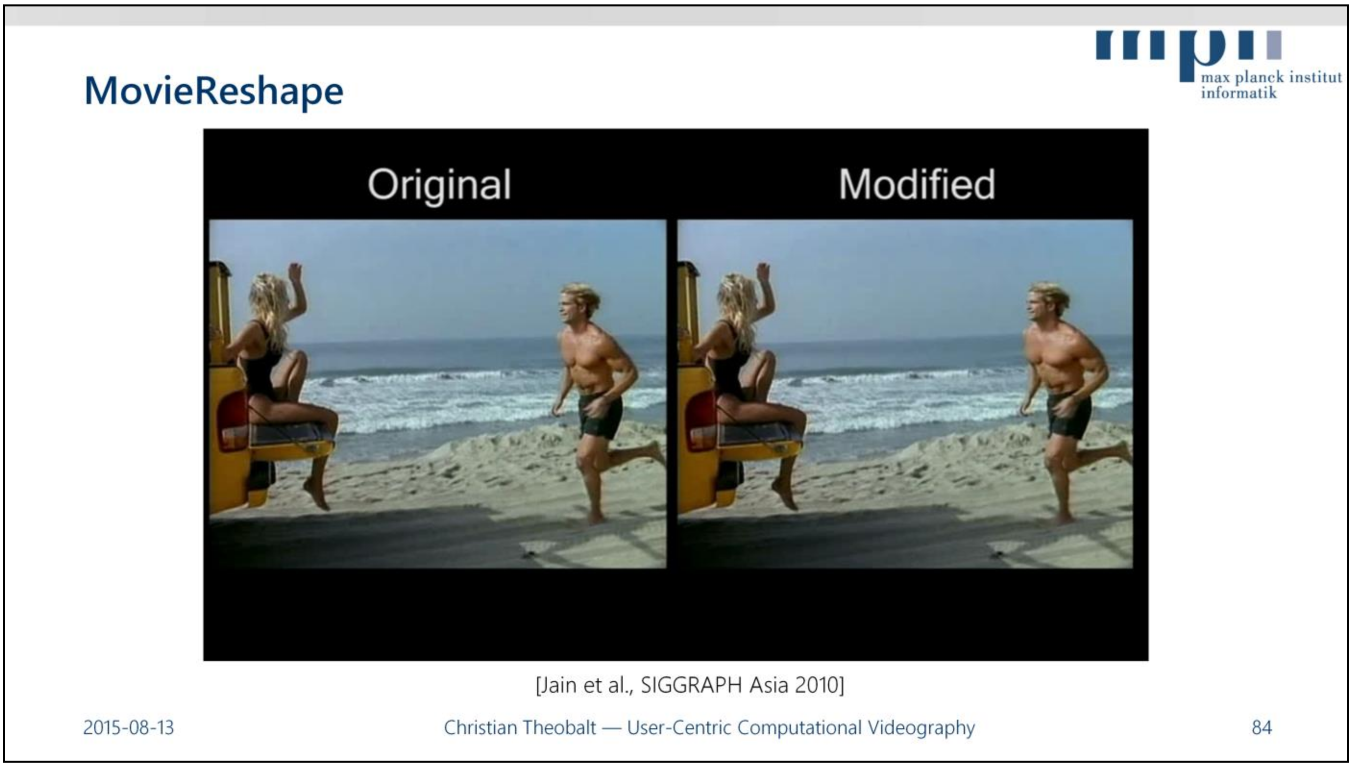
More recently, research made progress in enabling detailed dynamic shape capture of scenes recorded with less cameras, as well as in enabling dynamic shape reconstruction in less controlled environments where there are many elements in the scene, backgrounds are dynamic, and lighting can be uncontrolled and changing. Some works successfully capture dynamic shape geometry outdoors. In the following, we review example works.

In one line of work, progress was made possible by combining template-based dynamic shape capture from multi-view camera systems with an inverse rendering approach. Inverse rendering estimates the scene illumination and surface reflectance of shapes in general scenes by optimizing a model-to-image similarity measure. Once illumination and reflectance are known, it is feasible to also refine the surface geometry to extract microscale surface detail via shape-from-shading under uncontrolled lighting [Wu et al. 2011]. The additional shading-constraints can also be used to formulate the pose tracking energy functional of template-based performance capture approaches in a more robust way [Wu et al. 2012]. While the aforementioned works regard all surfaces as Lambertian surfaces during inverse rendering, it is also feasible to extract more advanced BRDF reflectance models from multi-view video captured under uncontrolled lighting, which yields relightable 3D videos [Li et al. 2013]. The aforementioned approaches still rely on a handful of video streams as input. More recently, it has been shown that the combination of template tracking and inverse rendering for illumination, reflectance and detail capture also enables space-time coherent 4D reconstruction (3D + time) of human full body performances [Wu et al. 2013], as well as face performances [Valgaerts et al. 2012] from only a handheld stereo pair of cameras. Reconstruction under changing lighting conditions indoors and outdoors is also feasible [Wu et al. 2012].

The ability to capture dynamic scenes with a single pair of cameras already dramatically enlarges the application range of 4D reconstruction methods. However, in order to employ these techniques for temporally coherent model-based video manipulation, they would need to run from monocular video footage only. 3D and 4D reconstruction from monocular footage is a highly challenging problem that is – in general – highly underconstrained. However, for monocular videos showing specific classes of objects, recent methods have shown the feasibility of high-quality 3D and 4D reconstruction since one can rely more strongly on prior information constraining the space of plausible reconstructions. In particular, the area of monocular face performance capture has recently progressed quite rapidly. Recent approaches typically rely on some form of parametric face model that is fitted to the images by solving an alignment optimization problem [Garrido et al. 2013, Shi et al. 2014, Ichim et al. 2015], by mesh template tracking [Suwajanakorn et al., 2014], or by applying a learned regression model for general face shape and expression [Cao et al. 2014], as well as face detail [Cao et al. 2015]. Many of these face capture approaches also employ on an inverse rendering approach to perform shape-from-shading-based geometry reconstruction under general uncontrolled conditions. Since this implies also the estimation of the incident lighting and face reflectance, one can use them for realistic model-based manipulation of face expression or appearance in video [Garrido et al. 2013, Shi et al., 2014, Garrido et al. 2015]. We will discuss examples for such model-based face video edits later in this section of the course.

For some other types of objects in a video, monocular dynamic shape capture is also feasible. Some approaches rely on a template of a deformable surface which is fitted to video by means of an analysis-by-synthesis approach [Salzmann et al. 2011, . Hilsmann et al. 2009]. By this means it is also feasible to replace the pattern on a t-shirt in monocular video. If the object in the scene exhibits a specific pattern, modification of its appearance is easier (such as the texture on piece of apparel [Scholz et al. 2006]); this corresponds to using a dense marker-pattern that simplifies tracking and reconstruction. However, this latter type of approaches requires active interference with the scene which is either not feasible or simply not possible in general video.

Anther category of methods employs non-rigid-structure-from-motion (NRSfM) techniques to capture the deforming shape and camera parameters from monocular video. They are applicable to more general classes of scenes, but the quality of reconstructed models is often not as high as with the aforementioned approaches that rely on stronger shape priors [Vincente et al. 2012, Agudo et al. 2014]. Also, simultaneous shape and appearance reconstruction, which would be needed for many video manipulation tasks, has not yet been shown in conjunction with NRSfM methods.



MovieReshape

[Jain et al., SIGGRAPH Asia 2010]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 84

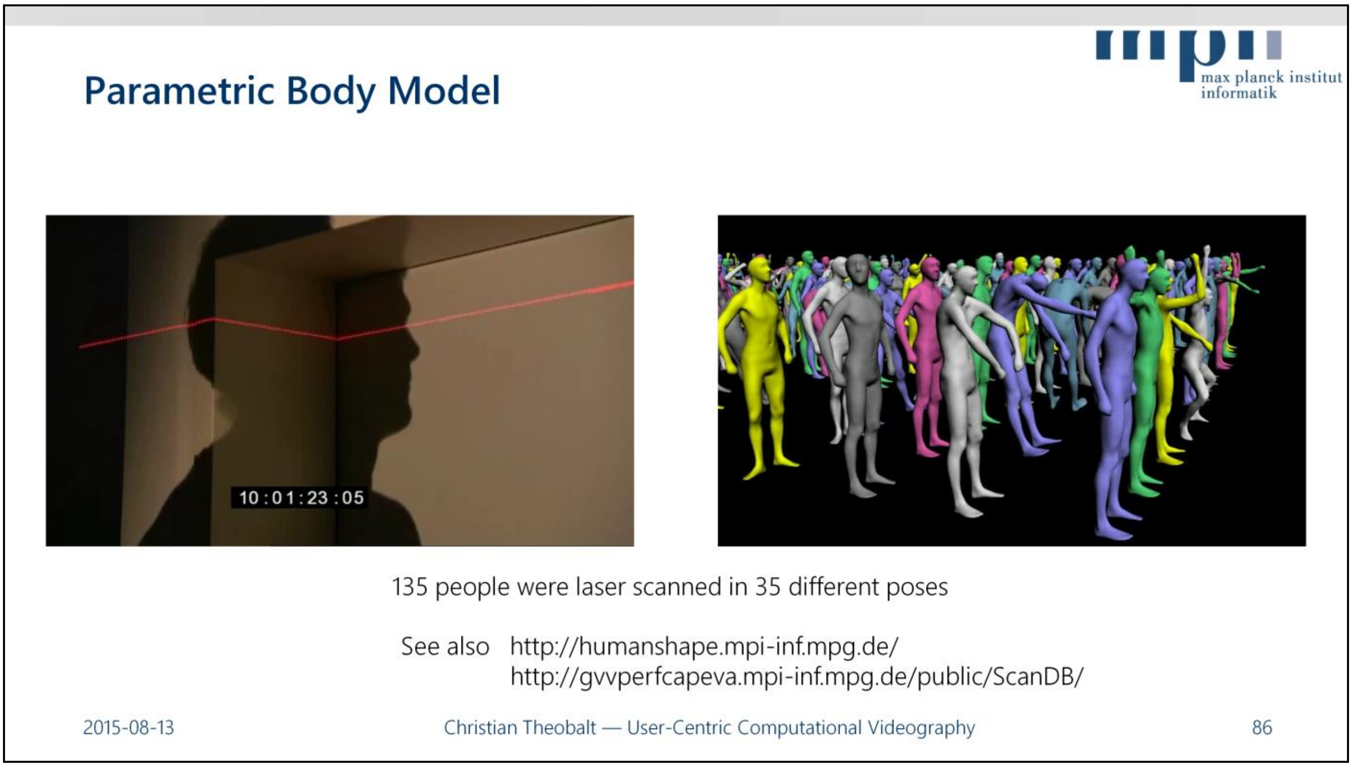
Let us now take a look at an example method for model-based video editing. The video above is an example result created with the MovieReshape algorithm from Jain et al. from SIGGRAPH Asia 2010. In the video above the physique of the actor shown in the original video (left) has been algorithmically altered to obtain the result on the right. In particular, the muscularity of the actor was enhanced. The modified result looks temporally coherent and realistically mimics the appearance of the original actor. In the following slides we look into its main concepts.

MovieReshape: Example Application

[excerpts from "The Lord of the Rings", and "Castaway"]

The effects that MovieReshape enables have many practical applications in professional VFX productions, as well as in postproduction of private videos. An example can be seen in the movie "the Lord of the Rings", where the Hobbits, who are played by human actors of normal size, are supposed to be much smaller than „real" humans shown alongside them in the same shot. In order to create this effect, it was necessary to resort to complex manual post-processing of frames, or the recording of scenes in several passes and later compositing. With MovieReshape, such effects could be achieved in a more efficient way.

Another example can be seen in the movie "Castaway" where the main actor, after spending a long time on a lonely island undergoes significant changes in his physique. A model-based postproduction tool may enable such visual changes in physique of an actor without the actor having to physically undergo these changes in reality.



Parametric Body Model

135 people were laser scanned in 35 different poses

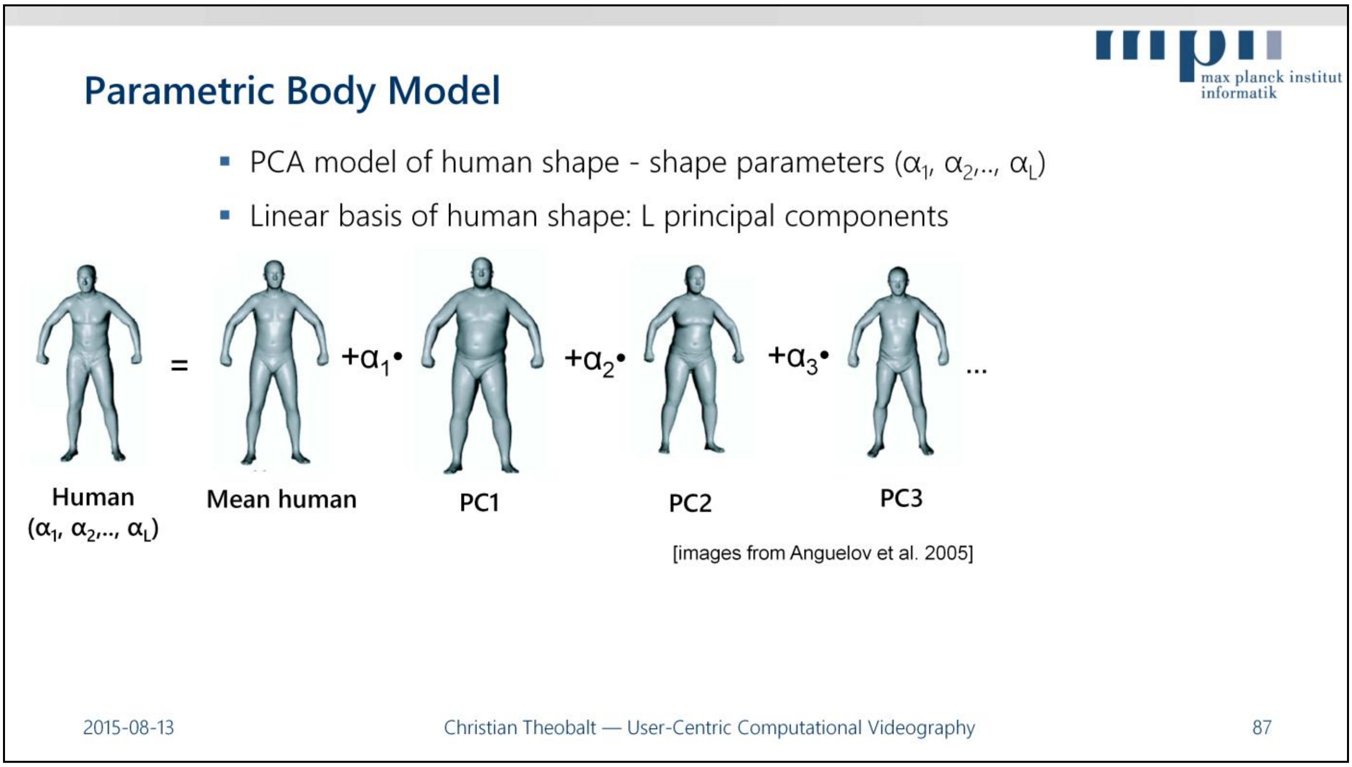
See also http://humanshape.mpi-inf.mpg.de/
http://gvvperfcapeva.mpi-inf.mpg.de/public/ScanDB/

2015-08-13 Christian Theobalt — User-Centric Computational Videography 86

One important component of MovieReshape is a statistical human body model that was created from a database of laser scanned humans. The model captures the variability of human body shape due to changes in pose and changes in human physique or gender. In other words, the model forms a prior that we will later use to capture the human pose and shape of an actor in the original video, and then to modify the human shape in a statistically plausible way.

Building this statistical body model is an off-line step that needs to be done once. It involves a variety of pre-processing steps, in particular registering a pre-defined shape template to the laser scans, such that all scans share the same underlying mesh structure. The development of algorithms for aligning meshes to scans is a research topic in itself and not the focus of this course. The original laser scans used for building the statistical model for MovieReshape can be found here: http://gvvperfcapeva.mpi-inf.mpg.de/public/ScanDB/.

Recently, we described in more detail a best practice pipeline to align template meshes to large scan databases in [Pischulin et al. 2015], and also make source code for scan alignment available. We also registered the entire CAESAR human body scan database, which features scans of close to 4000 individuals, to rebuild the MovieReshape body model form a larger database. We make the registered scans available under http://humanshape.mpi-inf.mpg.de/#overview.



Parametric Body Model

max planck institut informatik

- PCA model of human shape - shape parameters ($\alpha_1, \alpha_2, .., \alpha_L$)
- Linear basis of human shape: L principal components

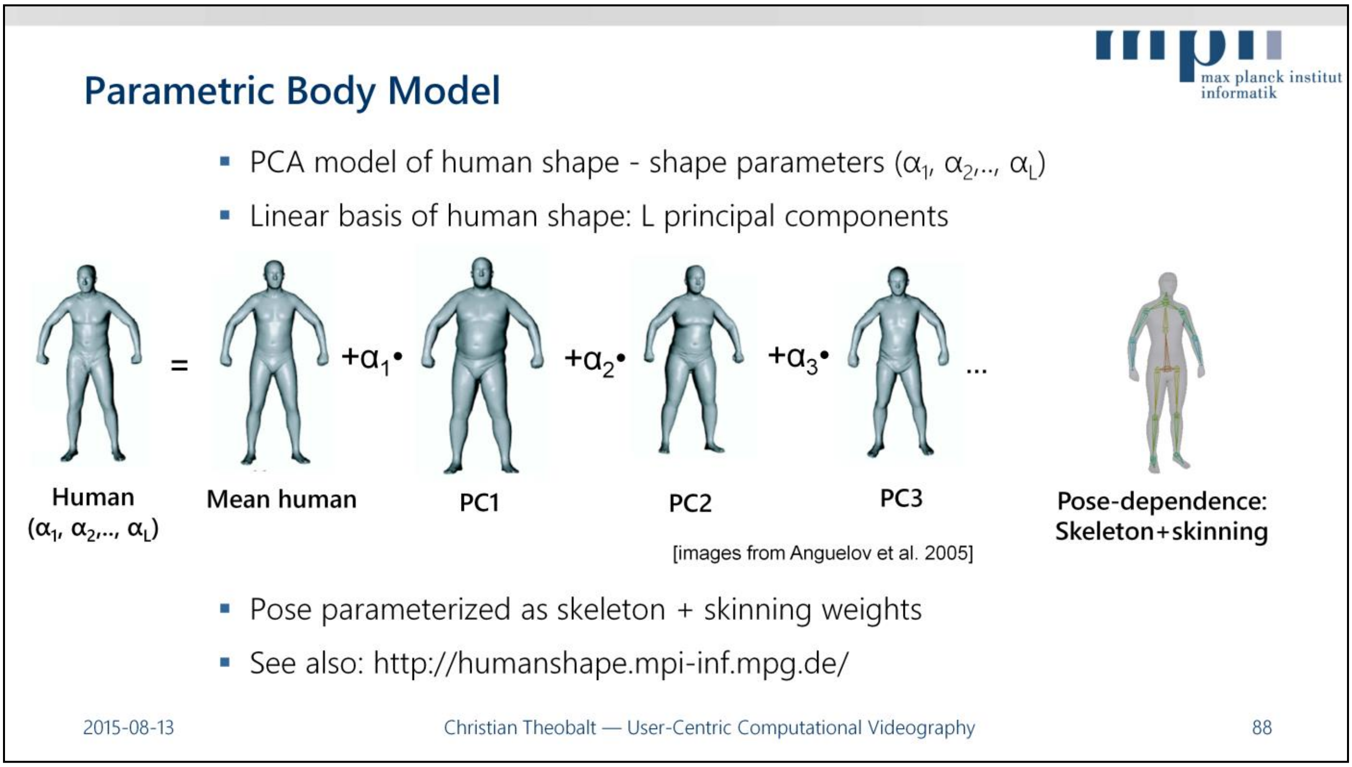
[images from Anguelov et al. 2005]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 87

Since the early SCAPE model [Anguelov et al. 2005] statistical models of human shape and pose have become valuable tools in computer vision and computer graphics. As explained on the previous slide, such models are learned from a database of registered body scans. The goal is to learn a representation of both the variation in human shape and human pose using a low dimensional parameter space.

The original SCAPE model and our simplified version of it parameterize the variations in body shape by means of a principal component analysis (PCA). PCA identifies the main directions of variation in a data distribution which form the so-called principal components. New body shapes in the PCA model can be created by a linear combination of the principal components and the mean body shape. In MovieReshape, we resort to the first 20 principal components for shape parameterization, which explain 97% of the total variation in body shape.

Pose in parametric human shape models is usually parameterized with a kinematic skeleton. Changes in skeleton pose entail surface variations that need to be parameterized. The SCAPE model by Anguelov et al. uses a rather complex parameterization of pose- and physique-dependent surface variations that necessitates a linear system solve to reconstruct the surface whenever the configuration has changed, this can create undesired computational overhead. In MovieReshape we use a simplified SCAPE model (see also [Pischulin et al. 2015]), in which pose-dependent shape variation is encoded with a standard surface skinning approach. The skeleton and skinning weights were designed once for the average human and are rescaled with changes in body shape. The skeleton dimensions scale with variations in body shape by expressing joint locations as weighted combinations of nearby surface vertices.



Parametric Body Model

- PCA model of human shape - shape parameters ($\alpha_1, \alpha_2, .., \alpha_L$)
- Linear basis of human shape: L principal components

[images from Anguelov et al. 2005]

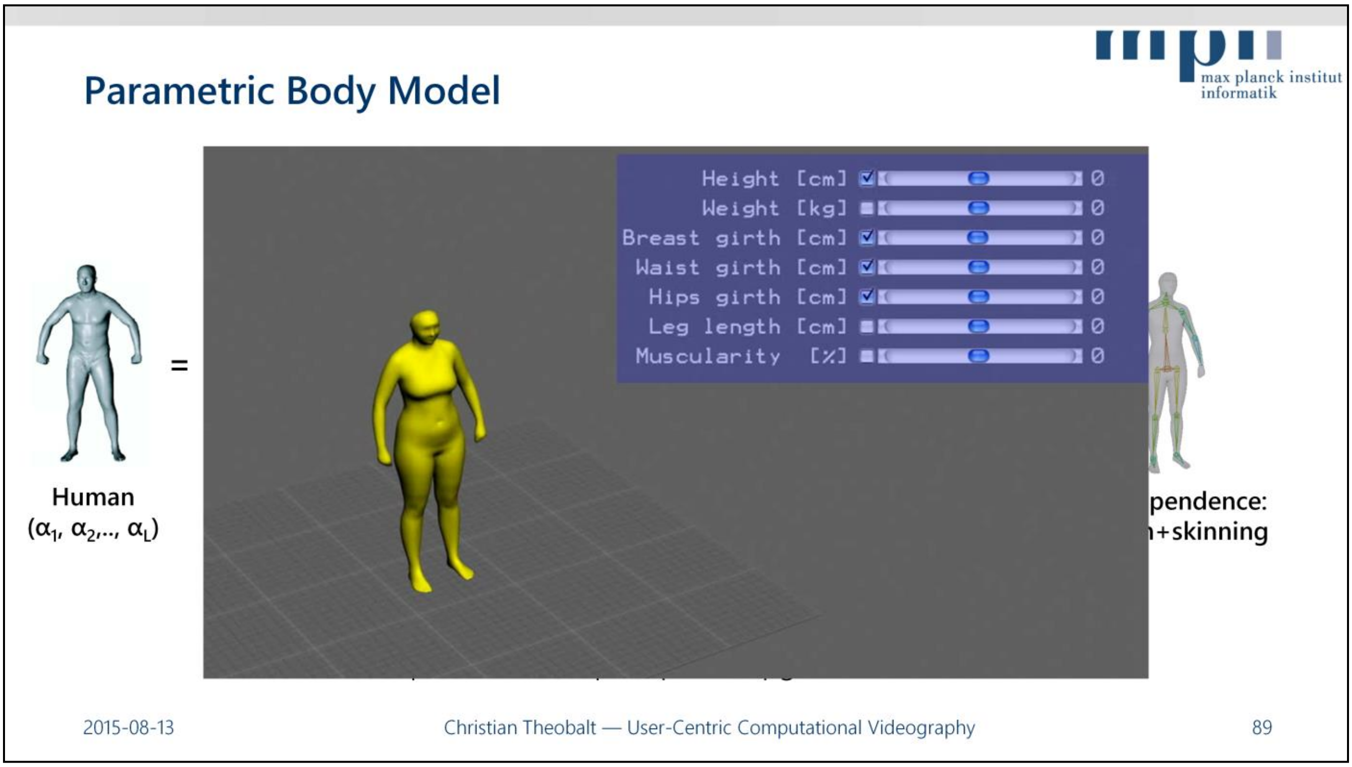
- Pose parameterized as skeleton + skinning weights
- See also: http://humanshape.mpi-inf.mpg.de/

2015-08-13 Christian Theobalt — User-Centric Computational Videography 88

Since the early SCAPE model [Anguelov et al. 2005] statistical models of human shape and pose have become valuable tools in computer vision and computer graphics. As explained on the previous slide, such models are learned from a database of registered body scans. The goal is to learn a representation of both the variation in human shape and human pose using a low dimensional parameter space.

The original SCAPE model and our simplified version of it parameterize the variations in body shape by means of a principal component analysis (PCA). PCA identifies the main directions of variation in a data distribution which form the so-called principal components. New body shapes in the PCA model can be created by a linear combination of the principal components and the mean body shape. In MovieReshape, we resort to the first 20 principal components for shape parameterization, which explain 97% of the total variation in body shape.

Pose in parametric human shape models is usually parameterized with a kinematic skeleton. Changes in skeleton pose entail surface variations that need to be parameterized. The SCAPE model by Anguelov et al. uses a rather complex parameterization of pose- and physique-dependent surface variations that necessitates a linear system solve to reconstruct the surface whenever the configuration has changed, this can create undesired computational overhead. In MovieReshape we use a simplified SCAPE model (see also [Pischulin et al. 2015]), in which pose-dependent shape variation is encoded with a standard surface skinning approach. The skeleton and skinning weights were designed once for the average human and are rescaled with changes in body shape. The skeleton dimensions scale with variations in body shape by expressing joint locations as weighted combinations of nearby surface vertices.



Parametric Body Model

max planck institut informatik

Human
($\alpha_1, \alpha_2, .., \alpha_L$)

=

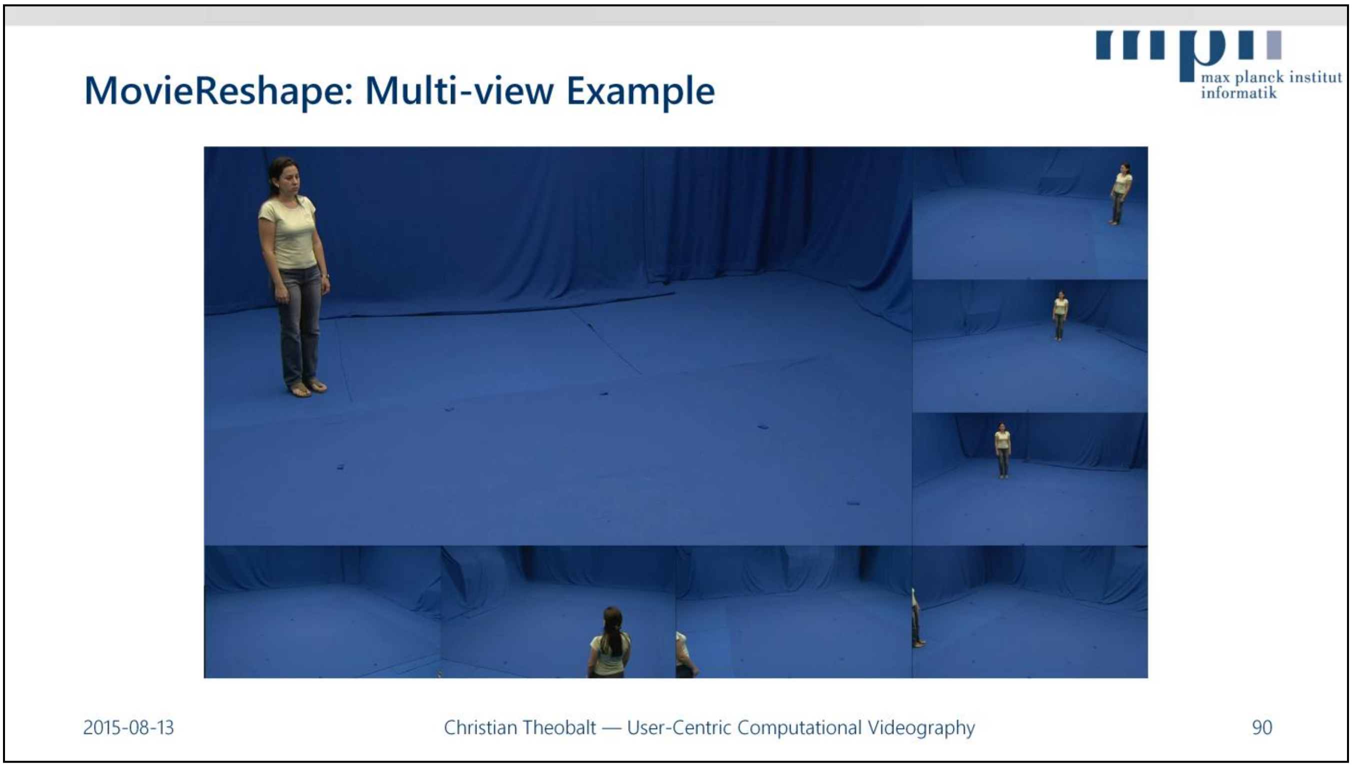
Height [cm] ☑ 0
Weight [kg] □ 0
Breast girth [cm] ☑ 0
Waist girth [cm] ☑ 0
Hips girth [cm] ☑ 0
Leg length [cm] □ 0
Muscularity [%] □ 0

pendence:
n+skinning

2015-08-13 Christian Theobalt — User-Centric Computational Videography 89

The automatically learned PCA dimensions of human shape variation do usually not have an intuitive semantic meaning. It is very likely that each PCA dimension encodes a combination of variations that a human would assign individual semantic labels to, such as changes in height, or changes in body weight. For the laser scanned subjects used for building the simplified SCAPE model of MovieReshape certain semantic attributes were recorded, such as height, weight, or a subjective muscularity level in percent. We can therefore learn a linear mapping from the space of attributes into the PCA space of body shapes. With this regression function it is feasible to modify the body shape parameters along semantically meaningful dimensions, as shown in the video above.

Video link: https://www.youtube.com/watch?feature=player_embedded&v=zXSj4pcl9Ao



MovieReshape: Multi-view Example

2015-08-13 Christian Theobalt — User-Centric Computational Videography 90

The video above shows the operation of MovieReshape using multi-view video footage captured in a specific studio with blue screen background. The steps will be explained in more detail on the next slides.

Video link: https://www.youtube.com/watch?feature=player_embedded&v=zXSj4pcl9Ao

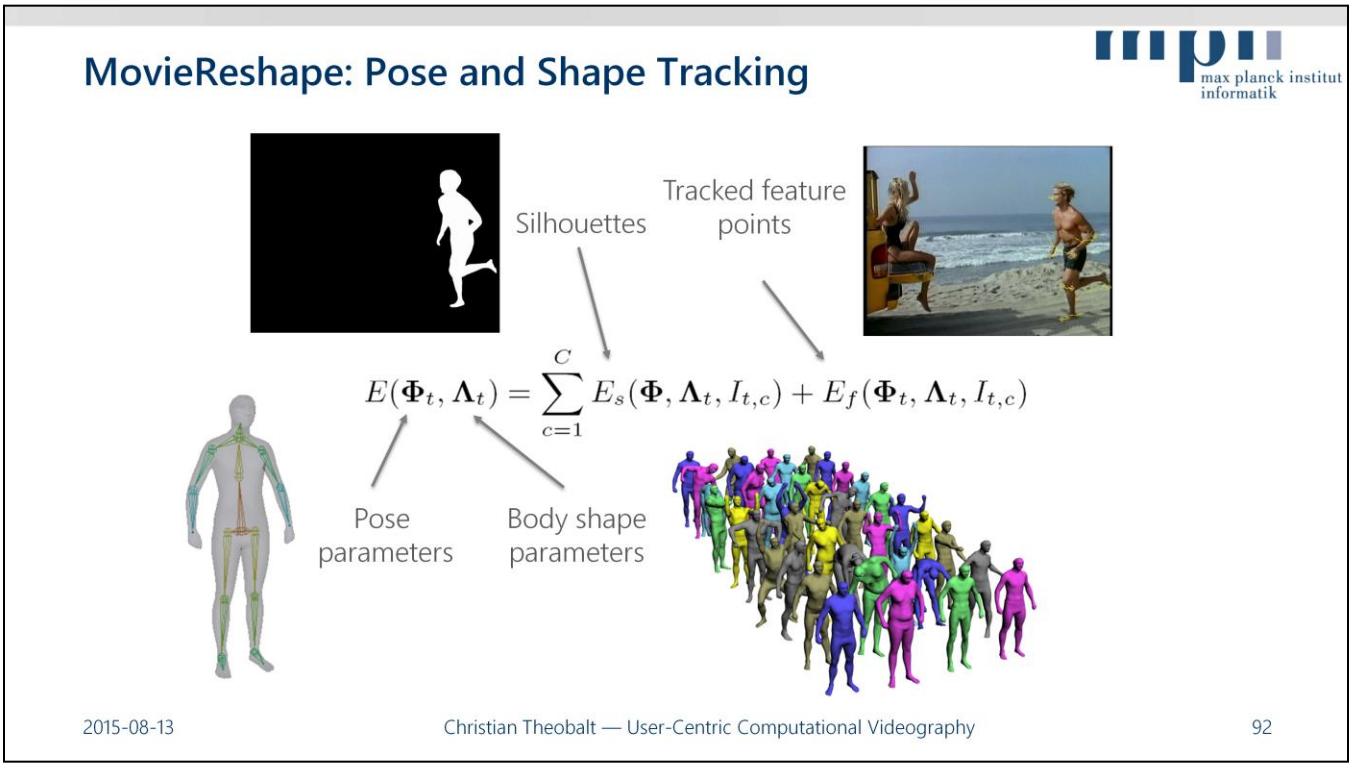
MovieReshape: Tracking and Reshaping in Multi-view Case

2015-08-13 Christian Theobalt — User-Centric Computational Videography 91

First, the pose and shape of the human actor are capture based on our statistical human body model. In case of multi-view video footage with perfect background segmentation, as shown here, this can be done automatically. Essentially an error function is minimized between the multi-view images and the reprojected parametric body model. This error / energy function encodes constraints from tracked features as well as silhouette boundaries. The exact details are also reviewed in Jain et al. [2009] and reviewed on the following slide.

After shape and pose at each time step were estimated, one can change the shape parameter of the model and keep the pose parameters the same. For instance, in the above example video the height of the body model is changed. Once the shape of the body model is modified it has to be propagated to the entire video of the actor. This is done by employing an image warping scheme based on moving least squares [Schaefer et al. 2006] using the projected reshaped model vertices as warping constraints in each video frame.

Video link: https://www.youtube.com/watch?feature=player_embedded&v=zXSj4pcl9Ao



MovieReshape: Pose and Shape Tracking

$$E(\mathbf{\Phi}_t, \mathbf{\Lambda}_t) = \sum_{c=1}^{C} E_s(\mathbf{\Phi}, \mathbf{\Lambda}_t, I_{t,c}) + E_f(\mathbf{\Phi}_t, \mathbf{\Lambda}_t, I_{t,c})$$

2015-08-13 Christian Theobalt — User-Centric Computational Videography 92

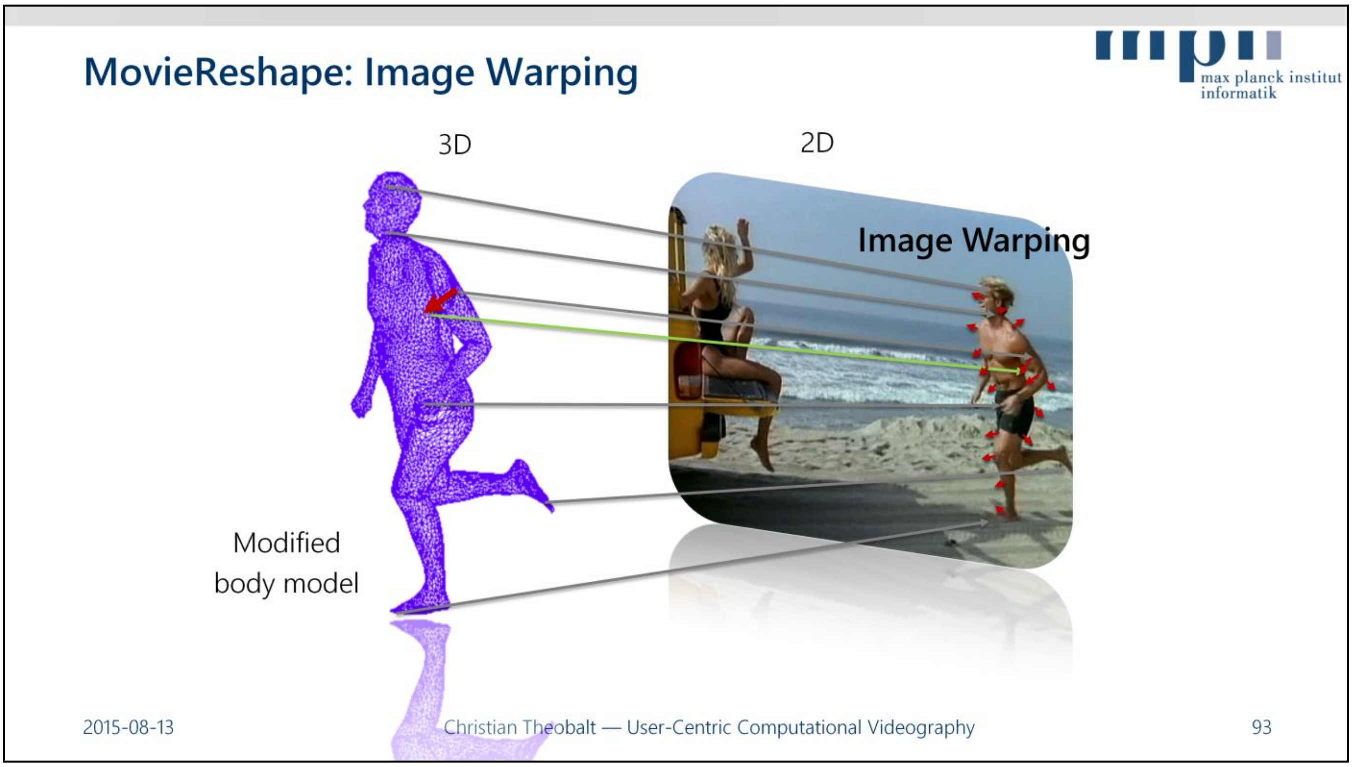
This slide illustrates the steps that are taken to capture the pose and shape of an actor in video based on the statistical human model we used in MovieReshape. The unknowns we need to recover are, firstly, the pose parameters for ach frame of video, i.e. the joint angles and the global pose of the body model. Second, we need to recover a fixed set of body shape parameters (PCA coefficients) for the entire video.

These unknown parameters are found by minimizing an error function. This error function encodes the alignment of our current body model with the person in each frame of video by means of two main types of constraints. The first type of constraint are silhouette images of the person in the foreground of the video. They can be computed by standard semi-automatic video segmentation algorithms as described earlier in this course. The first term in the above energy thus measures the misalignment of the projected body model and the silhouette of the person.

The second term measures the sum of distance between feature points tracked over time using KLT [Lucas et al. 1981], and the predicted featured locations from the rendered model. Automatic feature detection and tracking alone is often not sufficient, in particular if the input is a monocular video: Trajectories easily break or are interrupted due to self-occlusion, or feature points may not have been automatically found for body parts that are important but contain only moderate amounts of texture. MovieReshape therefore provides an interface in which the user can explicitly mark additional image points to be tracked, and in which broken trajectories can be linked.

The above energy is minimized by using the combined local and global pose optimization strategy of Gall et al. [2009]. Local pose optimization is extremely fast but may in some cases get stuck in incorrect local minima. Such pose errors could be prevented by running a full global pose optimization. However, global pose inference is prohibitively slow when performed on the entire pose and shape space. We therefore perform global pose optimization only for those sub-chains of the kinematic model, which are incorrectly fitted. Errors in the local optimization result manifest through a limb-specific fitting error that lies above a threshold. For global optimization, we utilize a particle filter variant.

In practice, we solve for pose and shape parameters in a hierarchical way. First, we solve for both shape and pose using only a subset of key frames of the video in which the actor shows a sufficient range pose and shape deformation. It turned out that in all our test sequences the first 20 frames form a suitable subset of frames. In this first optimization stage, we solely perform global pose and shape optimization and no local optimization. Thereafter, we keep the shape parameters fixed, and subsequently solve for the pose in all frame using the combined local and global optimization scheme.

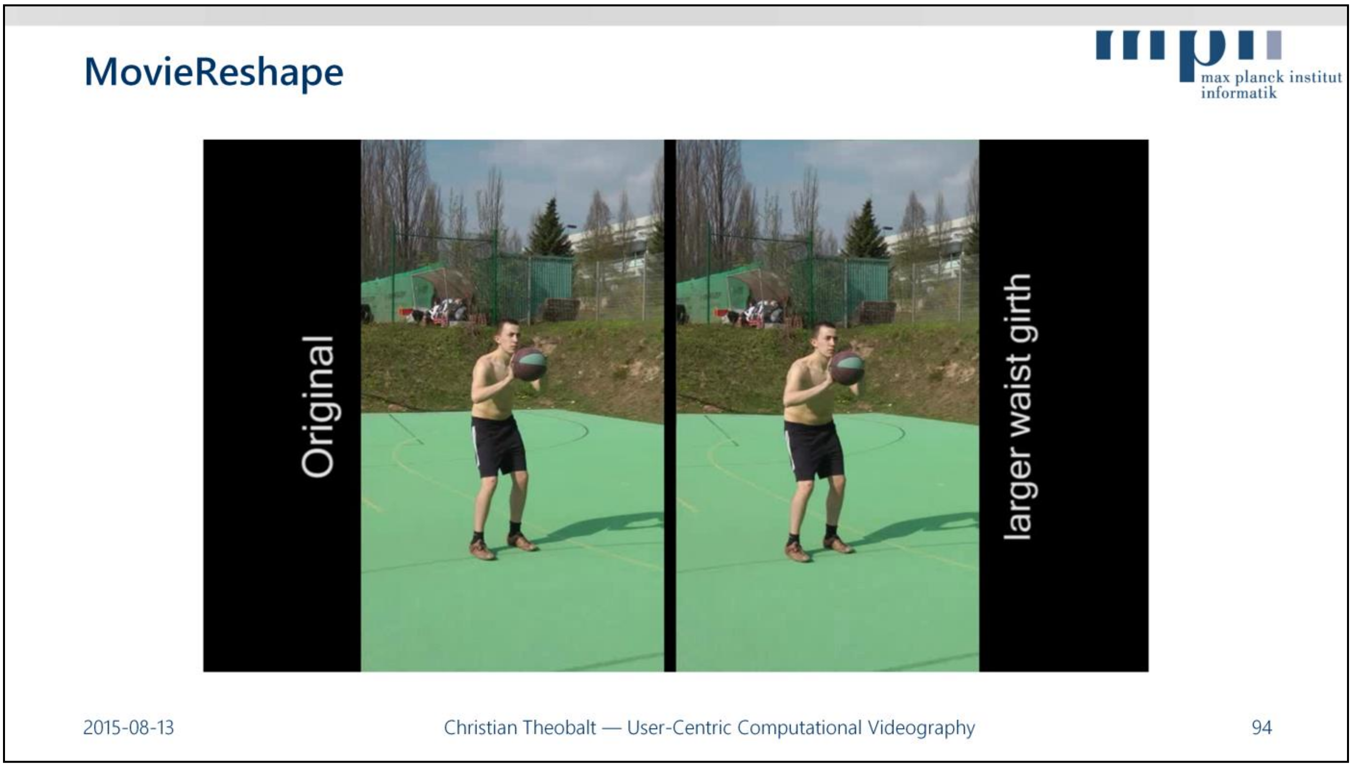


MovieReshape: Image Warping

2015-08-13 Christian Theobalt — User-Centric Computational Videography 93

Once the shape and pose of the actor in each frame is reconstructed, the shape parameters of the human can be modified using the MovieReshape interface. Changing the shape parameters of the model leads to 3D displacements of each vertex relative to the original shape. These 3D displacements translate to 2D displacements when projected into the image plane. Here, we assume that intrinsic calibration parameters of the video camera are estimated or given, such that this projection can be performed. The projected displacement vectors in each frame of video are then used as constraints in a moving least squares image warping approach [Schaefer et al. 2009].

Please note that each entire frame is warped, no segmentation of the foreground from background is used as this would imply the need to inpaint disoccluded regions. However, the downside is that the foreground warp may lead to unnatural deformations in a hallo around the warped person.



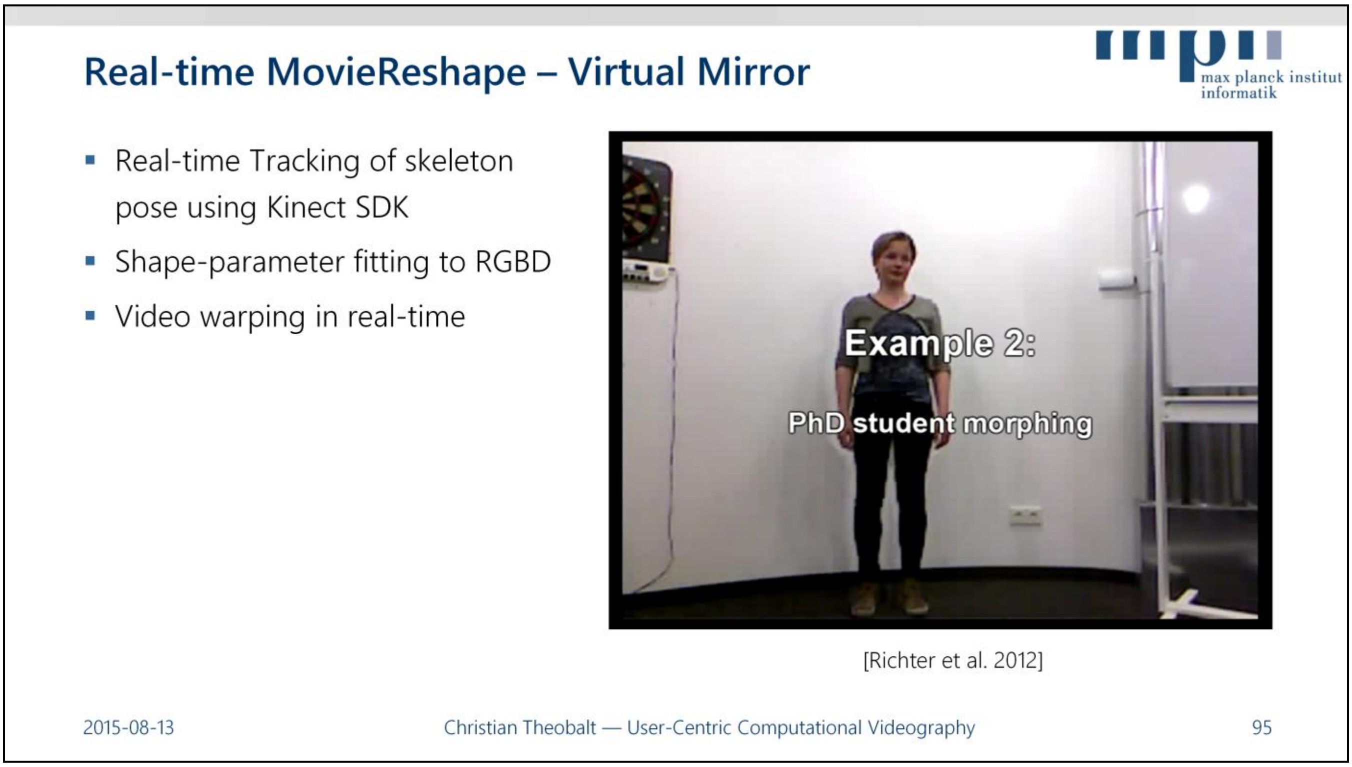
MovieReshape

2015-08-13 Christian Theobalt — User-Centric Computational Videography 94

Here is another result in which we realistically modified he waist girth of the actor. The final result looks very plausible and believably generates the appearance of the modified actor

Of course, the approach is subject to several limitations. Currently, monocular pose tracking is a semi-automatic process and user interaction is required. Also, the image warping may lead to unnatural deformations of the background in immediate vicinity of the person in the foreground. Currently, the approach only warps the appearance of each frame, but many shape changes may actually leads to shading differences in the image which are currently not modeled. Later on in this section, we will see approaches that can reproduce such realistic lshading effects after editing by resorting to inverse rendering methods.

Video link: https://www.youtube.com/watch?feature=player_embedded&v=zXSj4pcl9Ao



Real-time MovieReshape – Virtual Mirror

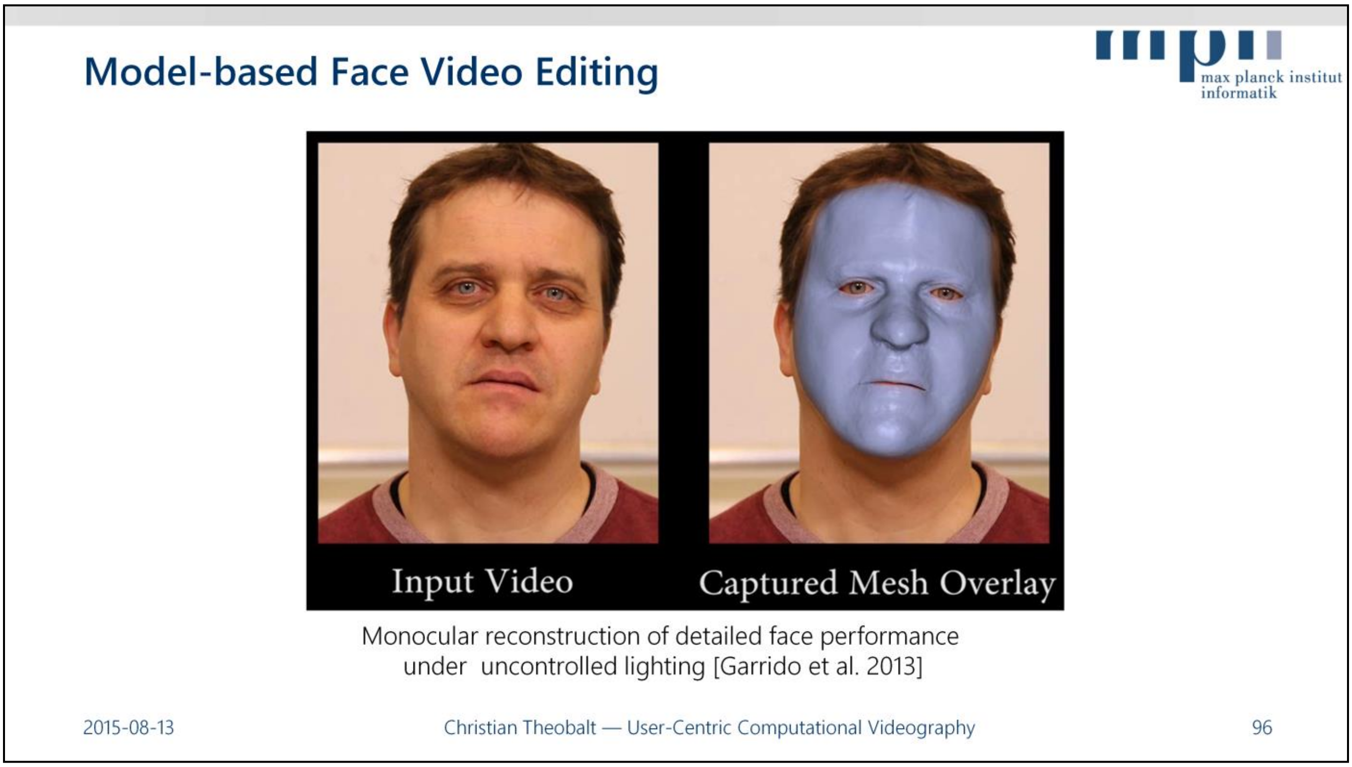
- Real-time Tracking of skeleton pose using Kinect SDK
- Shape-parameter fitting to RGBD
- Video warping in real-time

[Richter et al. 2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 95

As a follow-up, we developed a real-time version of MovieReshape that uses RGBD footage from a Kinect camera as input. In that version of the algorithm a person can look at herself being rendered with a modified body shape, thereby creating an experience akin to a virtual mirror [Richter et al. 2012].

Link to the video on this slide: http://gvv.mpi-inf.mpg.de/files/3DimPVT/ human_reshape.mp4



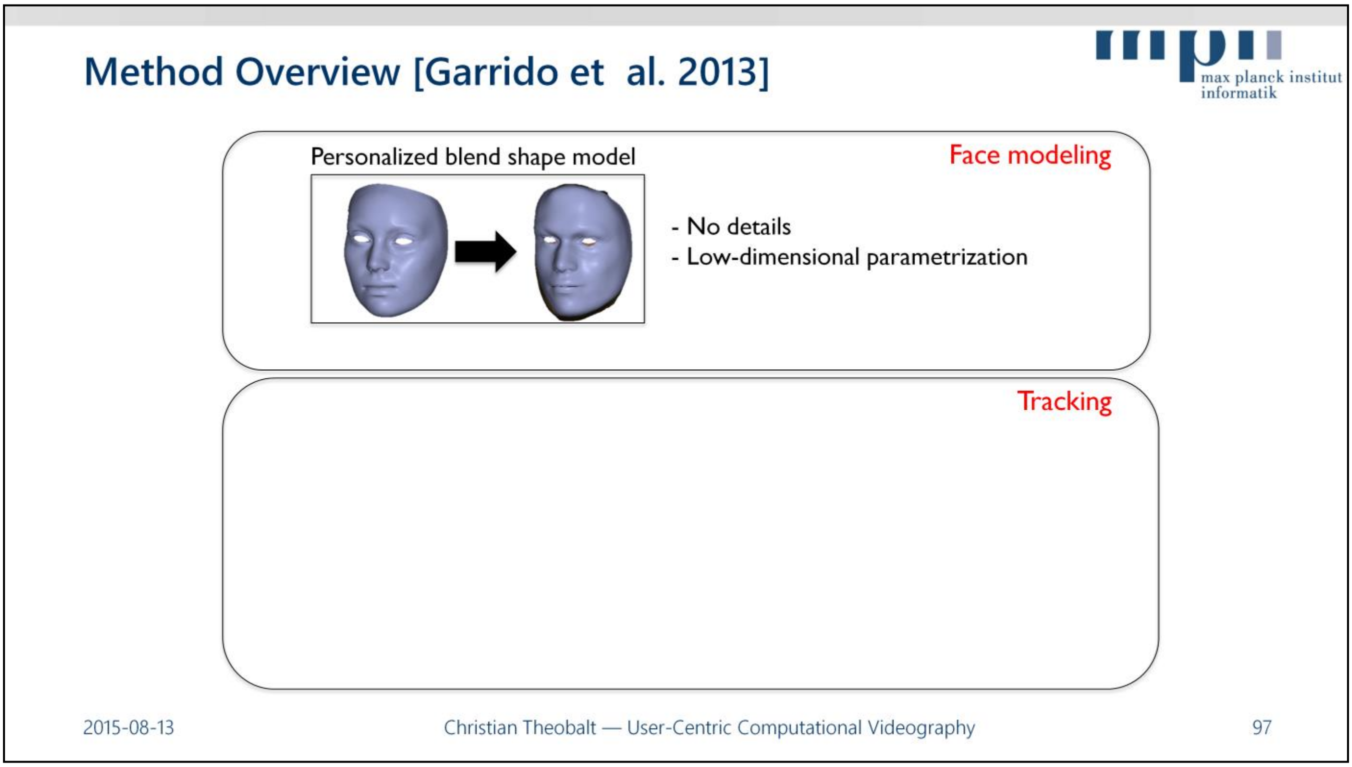
Model-based Face Video Editing

Monocular reconstruction of detailed face performance under uncontrolled lighting [Garrido et al. 2013]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 96

As discussed earlier, in particularly approaches enabling monocular performance capture of faces have made great strides ahead. Many of these approaches combine geometry tracking over time with some form of inverse rendering approach that produces an estimate of the scene illumination and reflectance, along with an estimate of a refined surface geometry. In the following, we will look in more detail into one of our approaches from this category of methods [Garrido et al. 2013].

Using a parametric face template, it reconstructs detailed space-time coherent scene geometry (see video), along with diffuse face albedo and incident lighting from monocular video. Lighting does not need to be engineered in a special way. In the following, we look in a bit more detail into the basics of this approach.



Method Overview [Garrido et al. 2013]

Face modeling

Personalized blend shape model

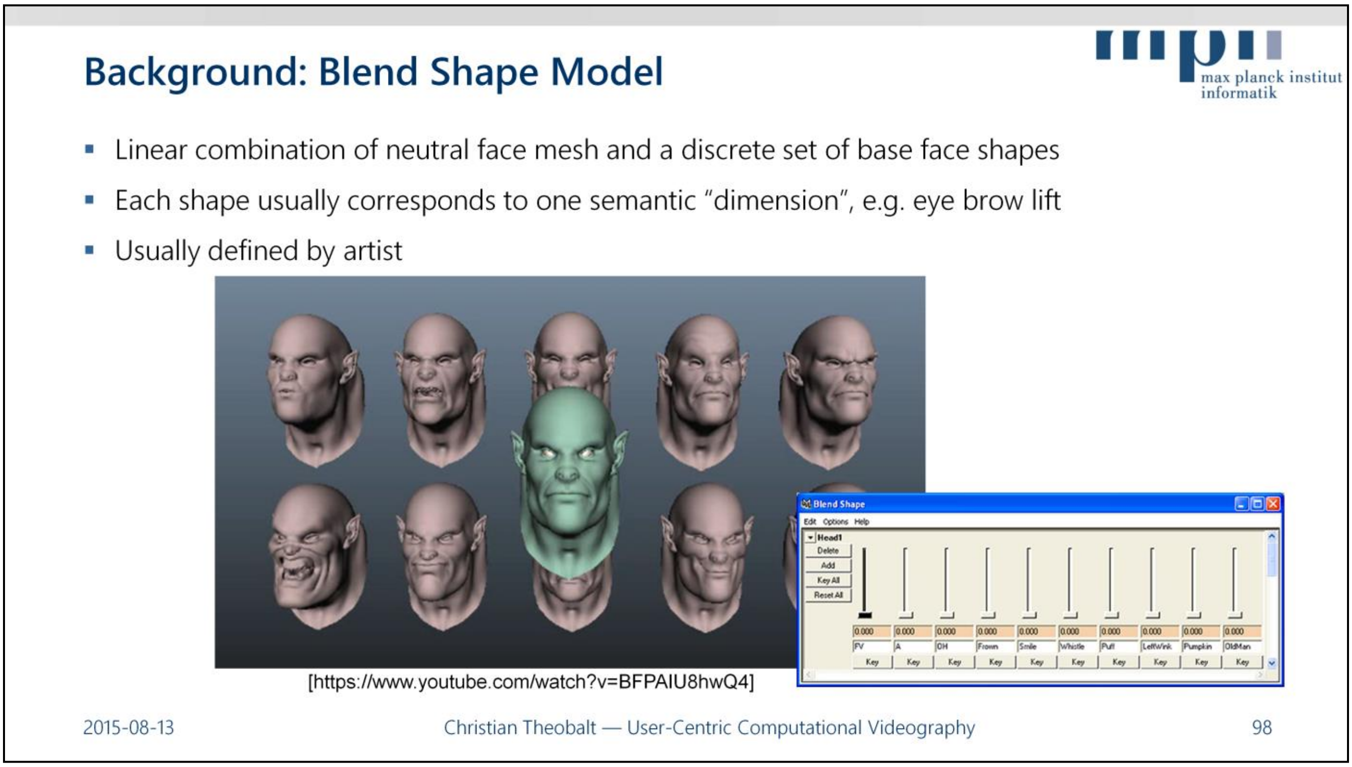
- No details
- Low-dimensional parametrization

Tracking

2015-08-13 Christian Theobalt — User-Centric Computational Videography 97

First, a parametric model of the face of the recorded actor is designed. This model captures the face geometry of the target actor (his identity), but also the space of possible face expressions on the basis a low-dimensional parameterization. In particular, we use a so-called blend shape model with 78 dimensions for face expressions (see next slide). The default blend shape model we use was designed for an actor with a specific face geometry. The blend shapes of that model therefore need to be personalized to the new target actor during pre-processing. To do so, we require a static 3D face scan of the actor in a neutral pose (e.g. from a stereo camera image) for preprocessing, and then we deform the default blend shapes to the new target model using deformation transfer.

Please note that this parametric model of the face only encodes low-to-medium scale shape detail. Fine scale face detail is later estimated in an additional step.



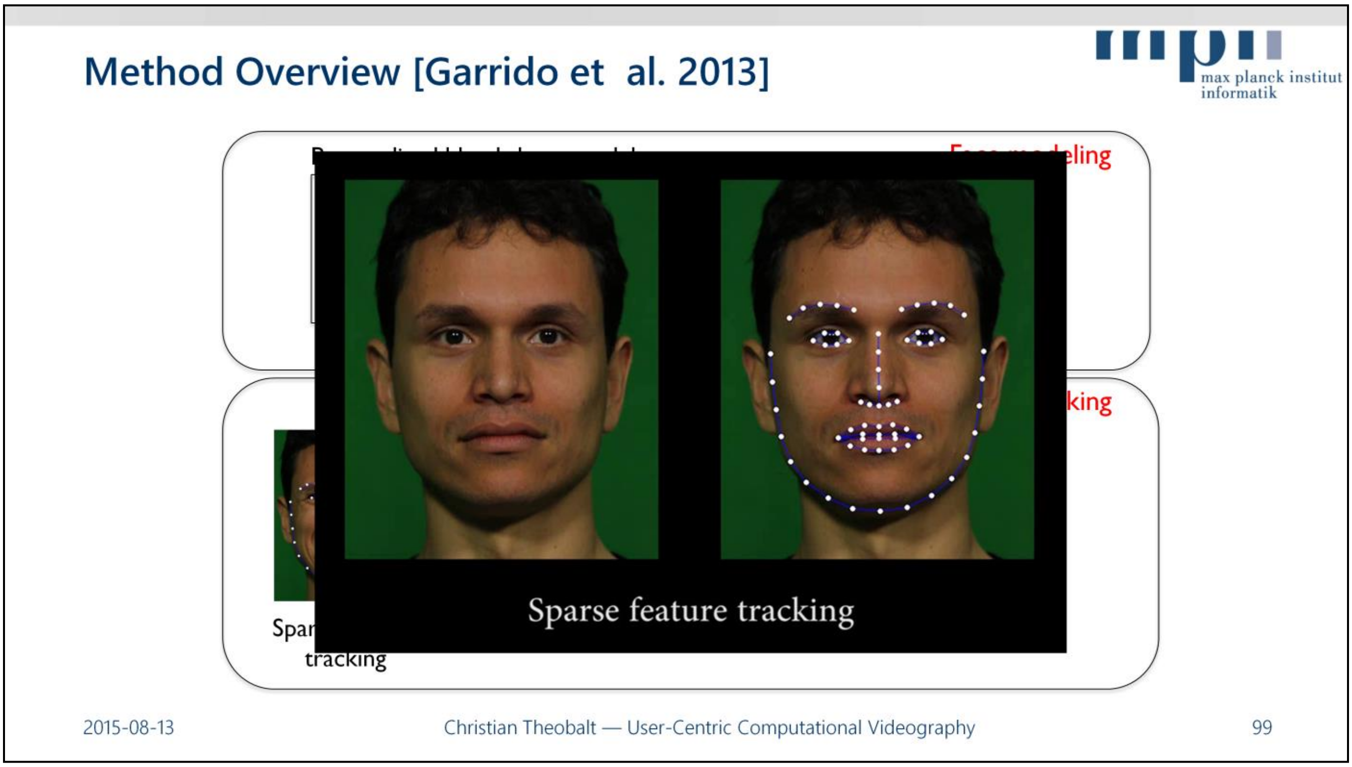
Background: Blend Shape Model

- Linear combination of neutral face mesh and a discrete set of base face shapes
- Each shape usually corresponds to one semantic “dimension”, e.g. eye brow lift
- Usually defined by artist

FV	A	OH	Frown	Smile	Whistle	Puff	LeftWink	Pumpkin	OldMan
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

[https://www.youtube.com/watch?v=BFPAIU8hwQ4]

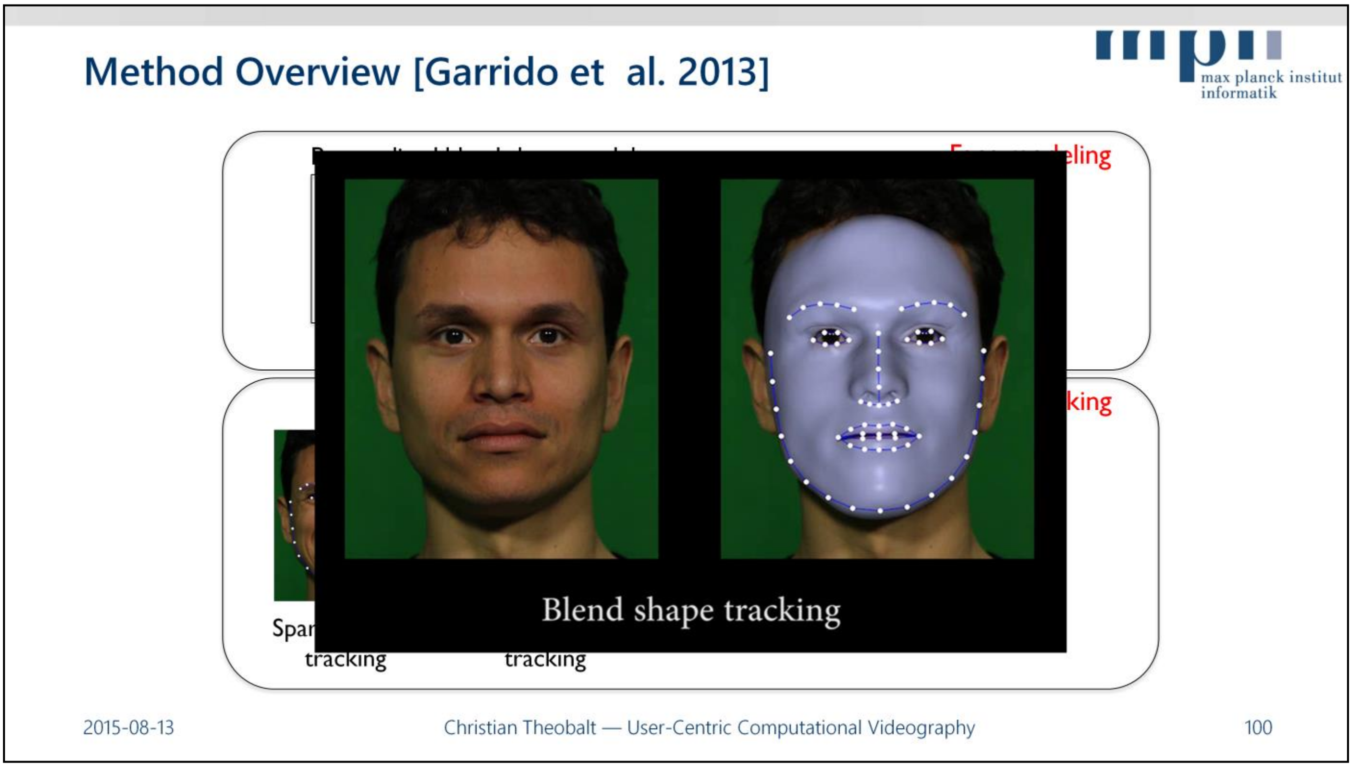
A blend shape model [Lewis et al. 2014] is a low dimensional model of face expressions widely used in computer animation. It comprises of a neutral face mesh and a set of facial expression meshes, the blend shapes. A new face expression is defined by a linear combination of the neutral face and the (difference of each blend shape) to the neutral face. Each blend shape’s influence is controlled by the controllable weight in the linear combination. When used for face performance capture, the blend shape weights of the recorded facial expressions need to be inferred from video.



Method Overview [Garrido et al. 2013]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 99

Given the personalized face model, the expressions and global pose of the face are tracked by solving an optimization problem. Given constraints extracted from the video frames, the 3D face shape and expression in each frame need to be determined such that the face model optimally overlaps with each video frame. The first set of constraints computed for tracking are sparse facial landmarks in each video frame, such as points on the lip contour, or the eye contour (see video). These landmarks are extracted using the approach by [Saragih et al., 2011].



Method Overview [Garrido et al. 2013]

eling

king

Spar

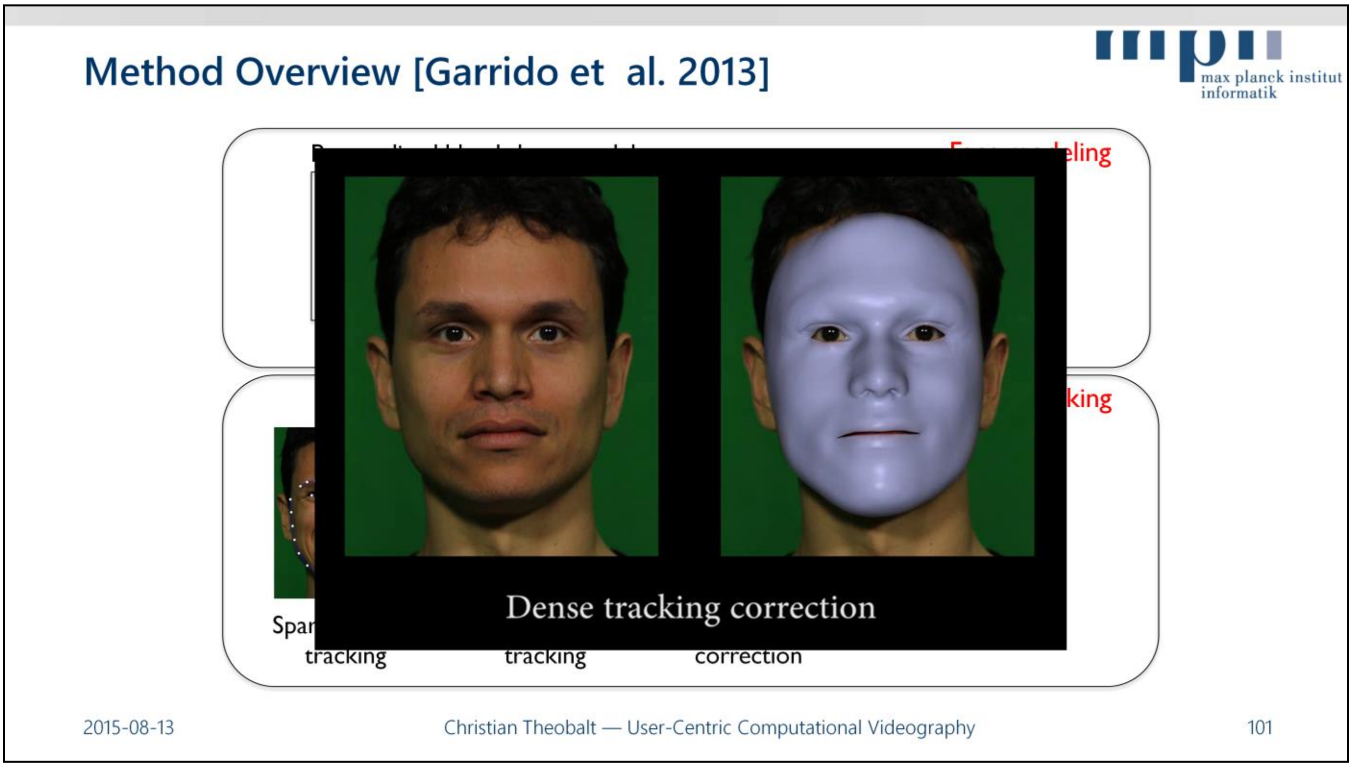
tracking

tracking

Blend shape tracking

2015-08-13 Christian Theobalt — User-Centric Computational Videography 100

The landmarks are used to estimate the rigid pose, as well as the optimal expression in terms of blend shape weights by solving a constrained linear optimization problem. Here we make use of the fact that correspondences were defined between specific 3D face model vertices and the detected landmarks in the images during pre-processing. The error metric minimized is thus the sum of 2D image position differences between tracked landmarks and projected landmark locations of the model.



Method Overview [Garrido et al. 2013]

eling

king

Spar

tracking

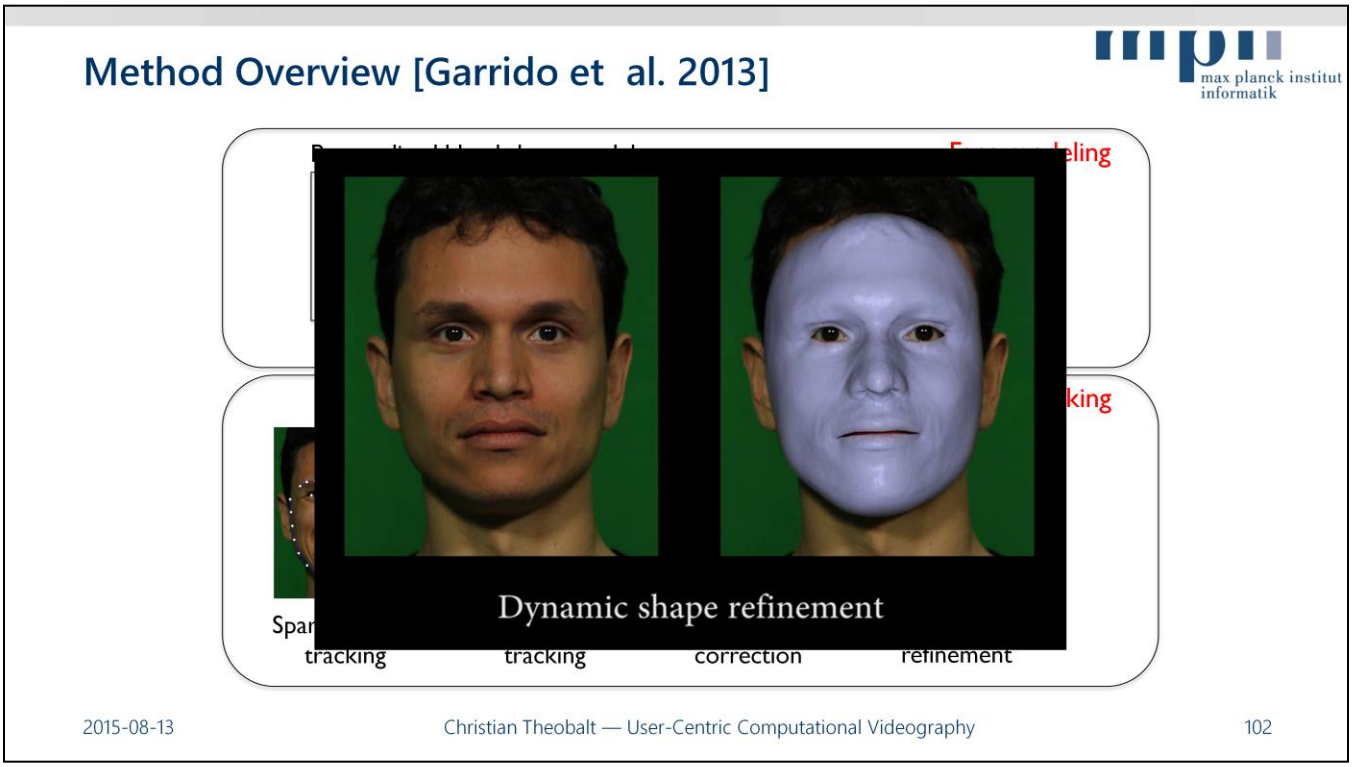
tracking

correction

Dense tracking correction

2015-08-13 Christian Theobalt — User-Centric Computational Videography 101

Since the blend shape space is not expressive enough to reproduce each face expression exactly, residual errors remain in the blend shape tracking, i.e. the model does not pixel-perfect align with the actor’s face in each video frame. In order to reduce these residual alignment errors the tracked mesh is densely aligned to the face image by employing a novel temporally-coherent method based on optic flow which. In other words, based on optic flow constraints and using a Laplacian non-rigid mesh deformation approach, the face model is warped to better match the image.



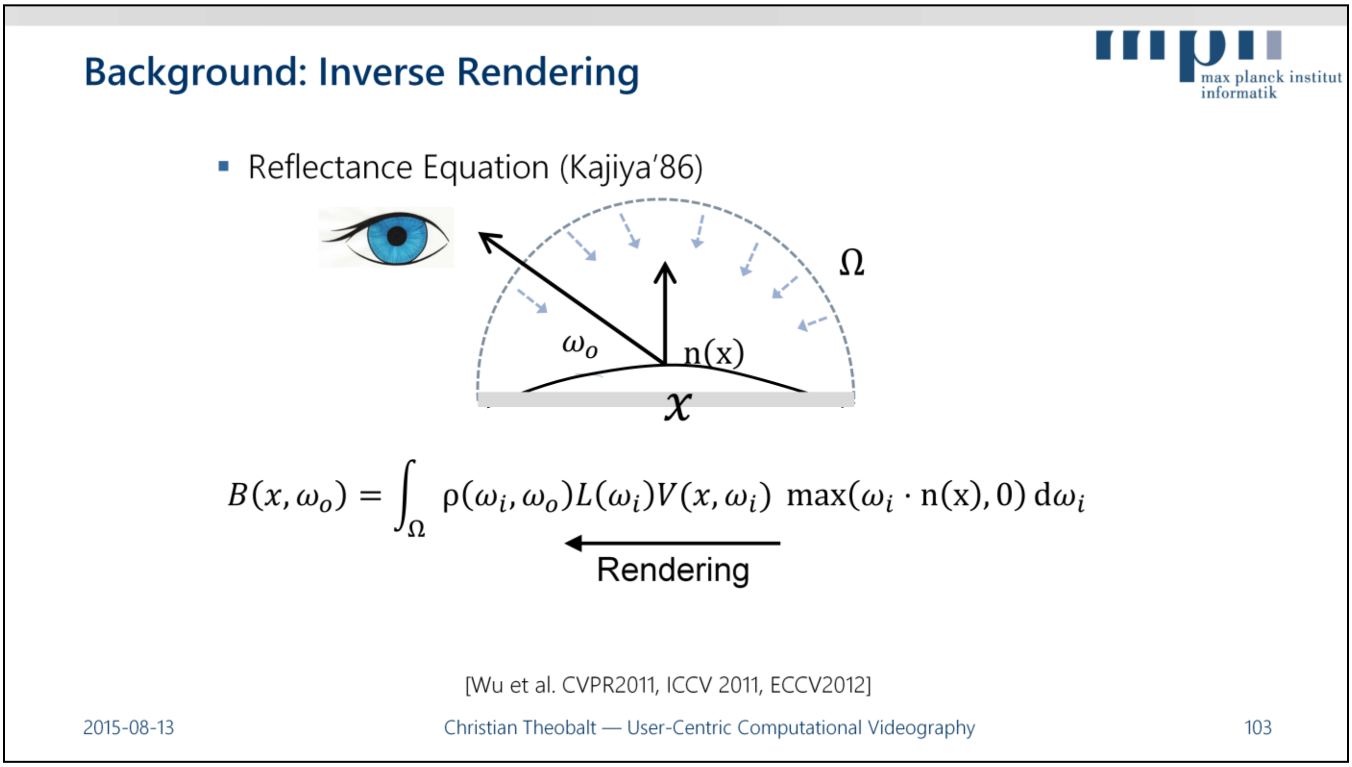
Method Overview [Garrido et al. 2013]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 102

By now, the face mesh aligns very well with the captured video frame in terms of global pose and expression. However, the face mesh is still very smooth and lacks fine expression detail, such as wrinkles and folds. The last step of face reconstruction therefore extracts a fine scale detail layer on top of the smooth face reconstruction obtained so far.

The core idea is to employ an inverse rendering approach. In a sense, the coarse deforming face mesh reconstructed so far is used as a generalized light probe. We assume a sparse set of diffuse face reflectance classes obtained in a clustering step, and assume that incident lighting is far away and can be expressed as an environment map parameterized in the spherical harmonics function basis. Given a few frames of video, the face albedo distribution and the incident lighting are first computed by minimizing an error energy functional that compares the rendered model to each frame of video. Once illumination and albedo are computed, the mesh geometry is subsequently optimized (deformed) such that rendered shading gradients correspond to observed shading gradients. Shape-refinement is performed in a space-time coherent manner, so refined geometry at each frame is regularized to be similar to the previous time step. The latter is also an energy minimization step, and altogether, the described inverse rendering approach thus corresponds to a shape-from-shading approach under unknown lighting.

The following slides take a side step and explain the basic concepts of this inverse rendering approach for shape-from-shading under general lighting.



Background: Inverse Rendering

- Reflectance Equation (Kajiya'86)

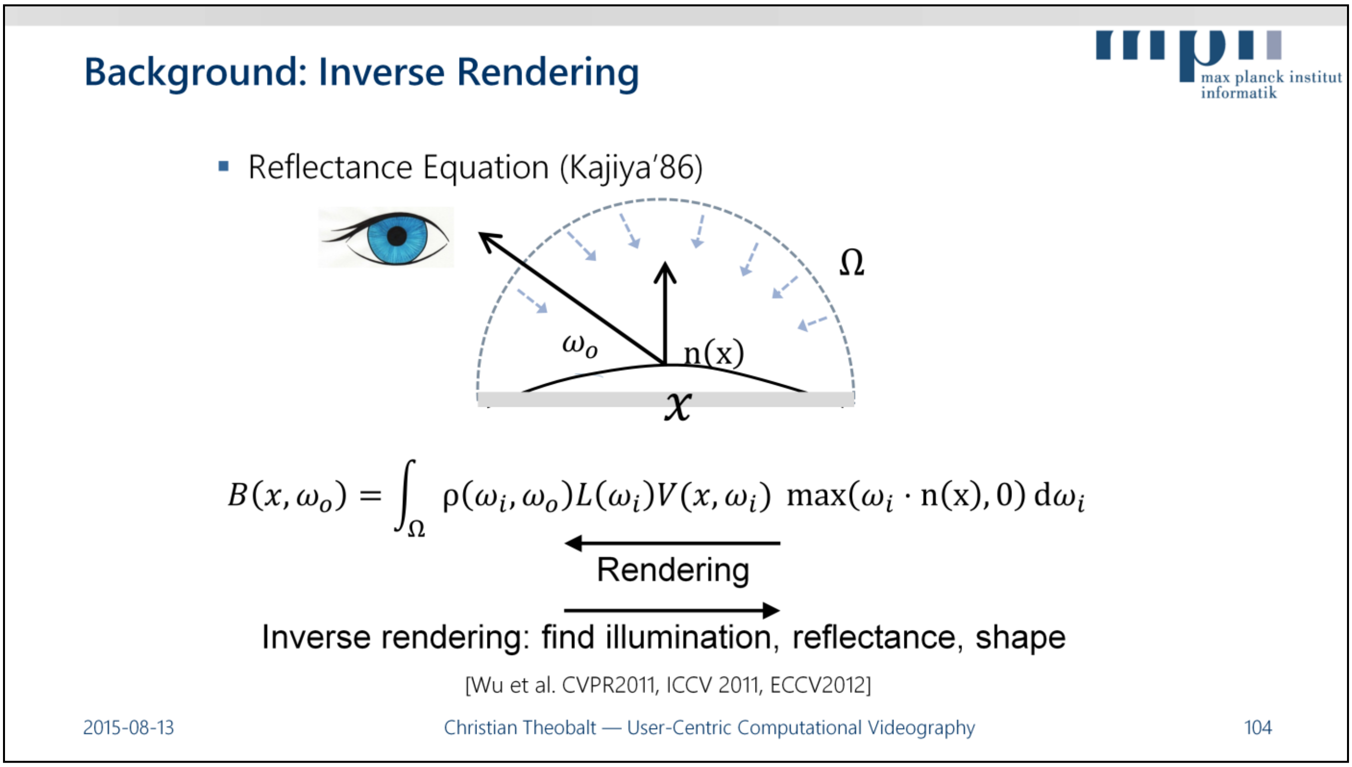
$$B(x,\omega_o) = \int_{\Omega} \rho(\omega_i,\omega_o) L(\omega_i) V(x,\omega_i)\ \max(\omega_i \cdot \mathrm{n}(x), 0)\, \mathrm{d}\omega_i$$

Rendering

[Wu et al. CVPR2011, ICCV 2011, ECCV2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 103

Let's briefly look at the idea of inverse rendering from the perspective of the rendering process (see also earlier slides on the reflectance equation). The reflectance equation [Kajiya 1986] describes the outgoing light from a point **x** with normal **n** towards the viewer in direction $\boldsymbol{\omega_o}$ if the point is illuminated with incident lighting **L** from directions $\boldsymbol{\Omega}$. The equation depends on the BRDF ρ of the point describing its reflectance, lighting **L,** the visibility function **V** towards the light, as well as the cosine of lighting and surface normal. In rendering we assume that all elements on the right hand side are given or can be computed, and then evaluate it to create a synthetic image of a scene.



Background: Inverse Rendering

- Reflectance Equation (Kajiya'86)

$$B(x,\omega_o) = \int_\Omega \rho(\omega_i,\omega_o) L(\omega_i) V(x,\omega_i)\ \max(\omega_i \cdot \mathrm{n(x)}, 0)\, \mathrm{d}\omega_i$$

$\longleftarrow$ Rendering

$\longrightarrow$

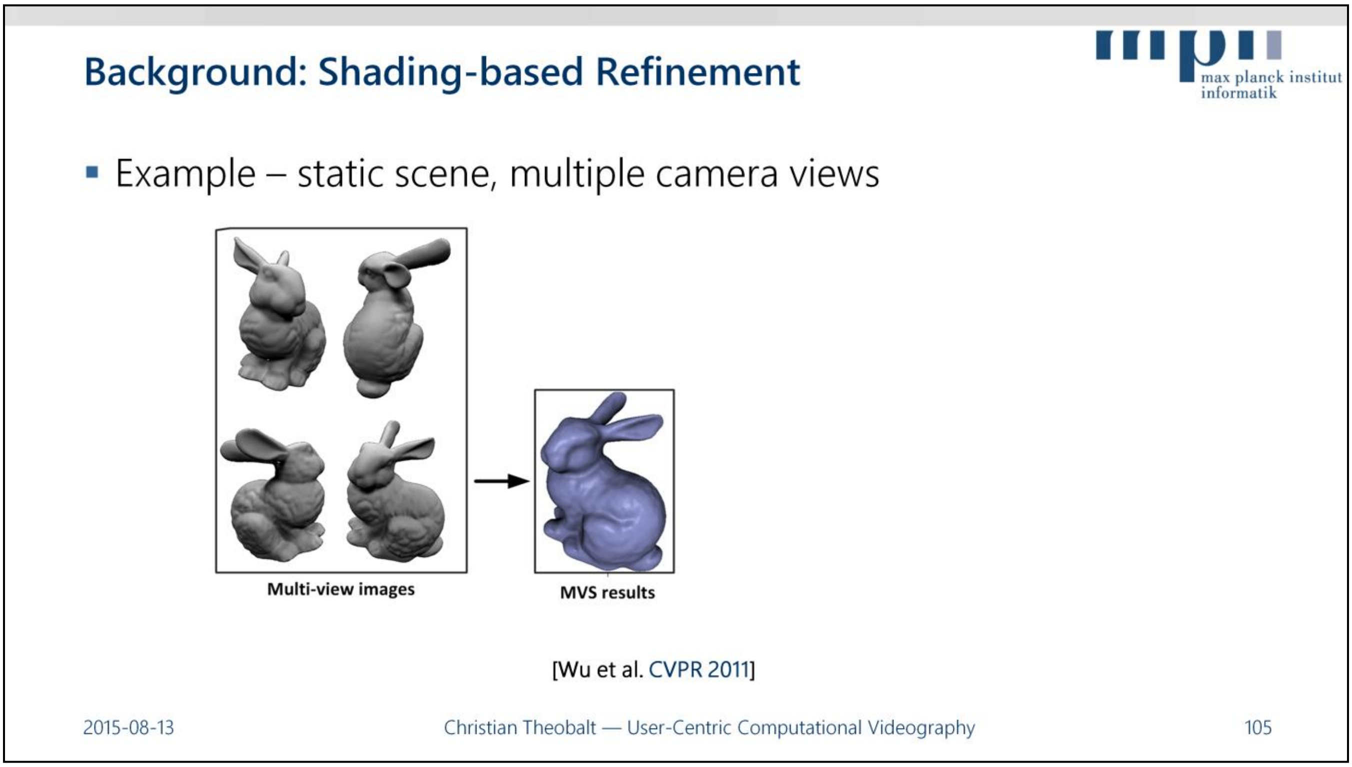
Inverse rendering: find illumination, reflectance, shape

[Wu et al. CVPR2011, ICCV 2011, ECCV2012]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 104

As the name suggest, inverse rendering is the opposite problem – given an image or several images of a scene, the entities on the right hands side need to be estimated, in particular the reflectance ρ and incident lighting **L**, as well as the shape of the object itself.

In the following, we assume that we deal with Lambertian objects, such that the BRDF is a constant diffuse albedo.



Background: Shading-based Refinement

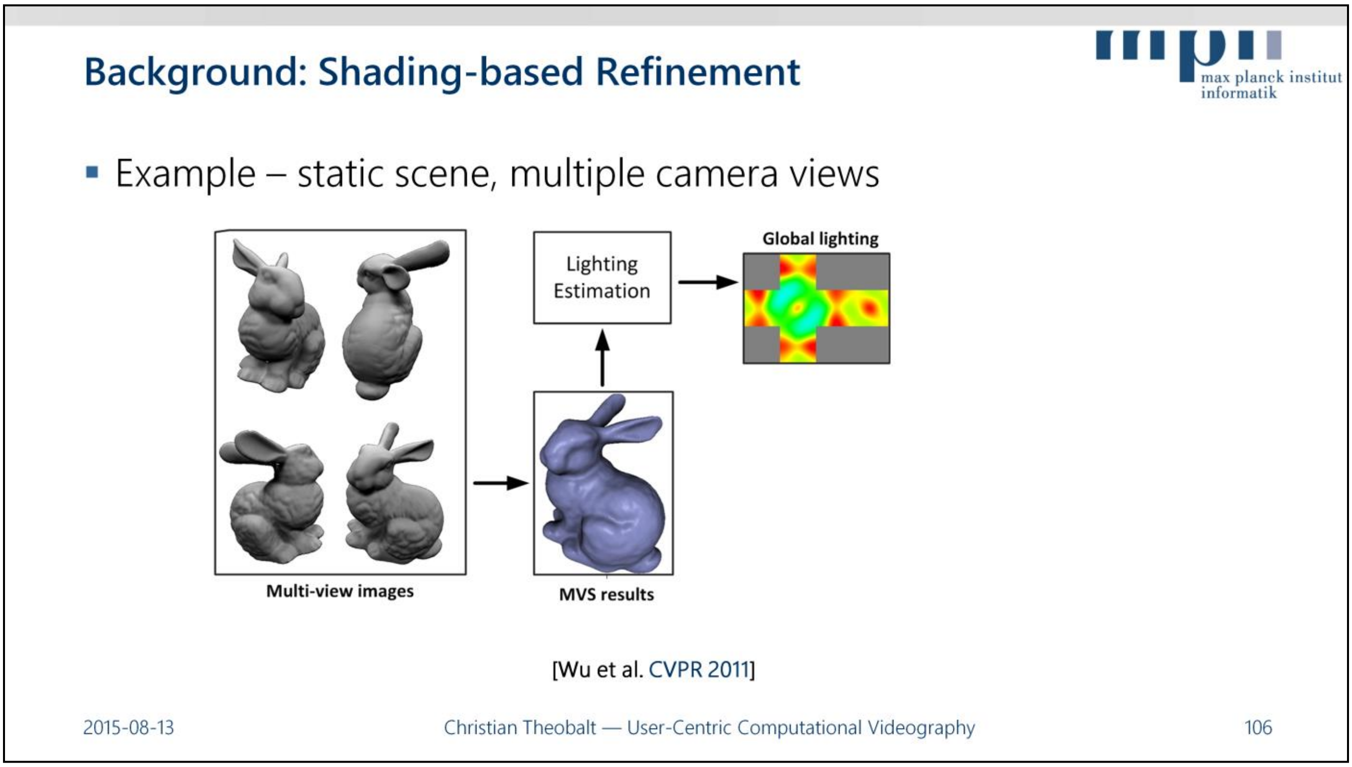
- Example – static scene, multiple camera views

[Wu et al. CVPR 2011]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 105

The following slides illustrate the steps taken for shape-from-shading-based geometry refinement (or short shading-based refinement) under general uncontrolled lighting. For explaining the process we follow our approach for multi-view shading-based refinement as described in [Wu et al. 2011]. This method assumes a constant diffuse albedo distribution on the object, and assumes that the object to be reconstructed is static and recorded from multiple camera views. This approach has later on be refined in several ways to handle monocular views, a sparse set of albedo distributions on the surface, as well as space-time coherent shading-based refinement in dynamic scenes. In other words, the explanation is based on a slightly different setting than in monocular face reconstruction. However, the basic concepts remain the same in both settings.

Let us assume that the input to our approach is a set of (geometrically and photometrically) calibrated images of an object under general lighting. Also, let’s assume that an initial shape estimate of the object is given, e.g. from stereo reconstruction or a template-based fitting approach as in our performance capture setting. The initial model can be a coarse representation of the true shape of the object.

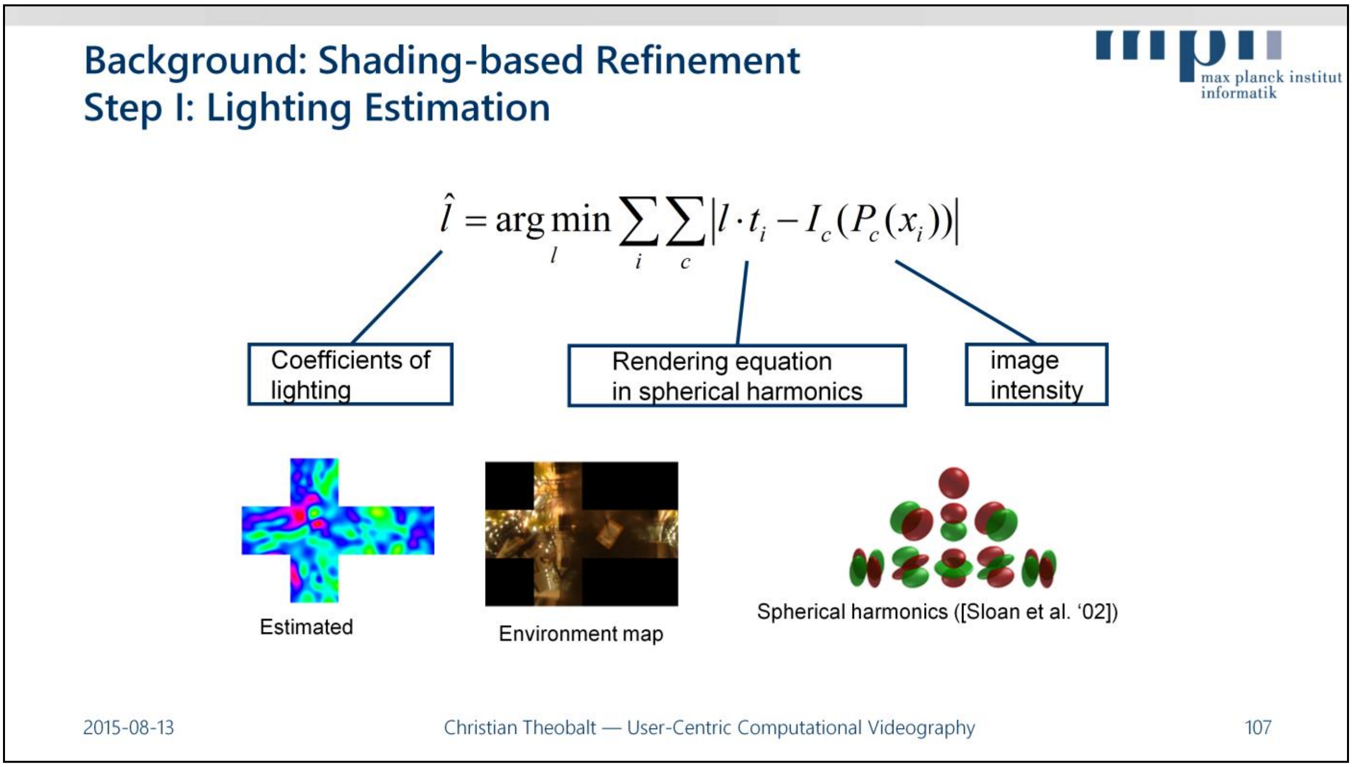


Background: Shading-based Refinement

- Example – static scene, multiple camera views

[Wu et al. CVPR 2011]

In a first step, we use the coarse initial object model as a generalized light probe in the scene and estimate the incident scene illumination. The next slide looks at this first step in a bit more detail.



Background: Shading-based Refinement
Step I: Lighting Estimation

$$\hat{l} = \arg\min_{l} \sum_{i} \sum_{c} \left| l \cdot t_i - I_c(P_c(x_i)) \right|$$

Coefficients of lighting

Rendering equation in spherical harmonics

image intensity

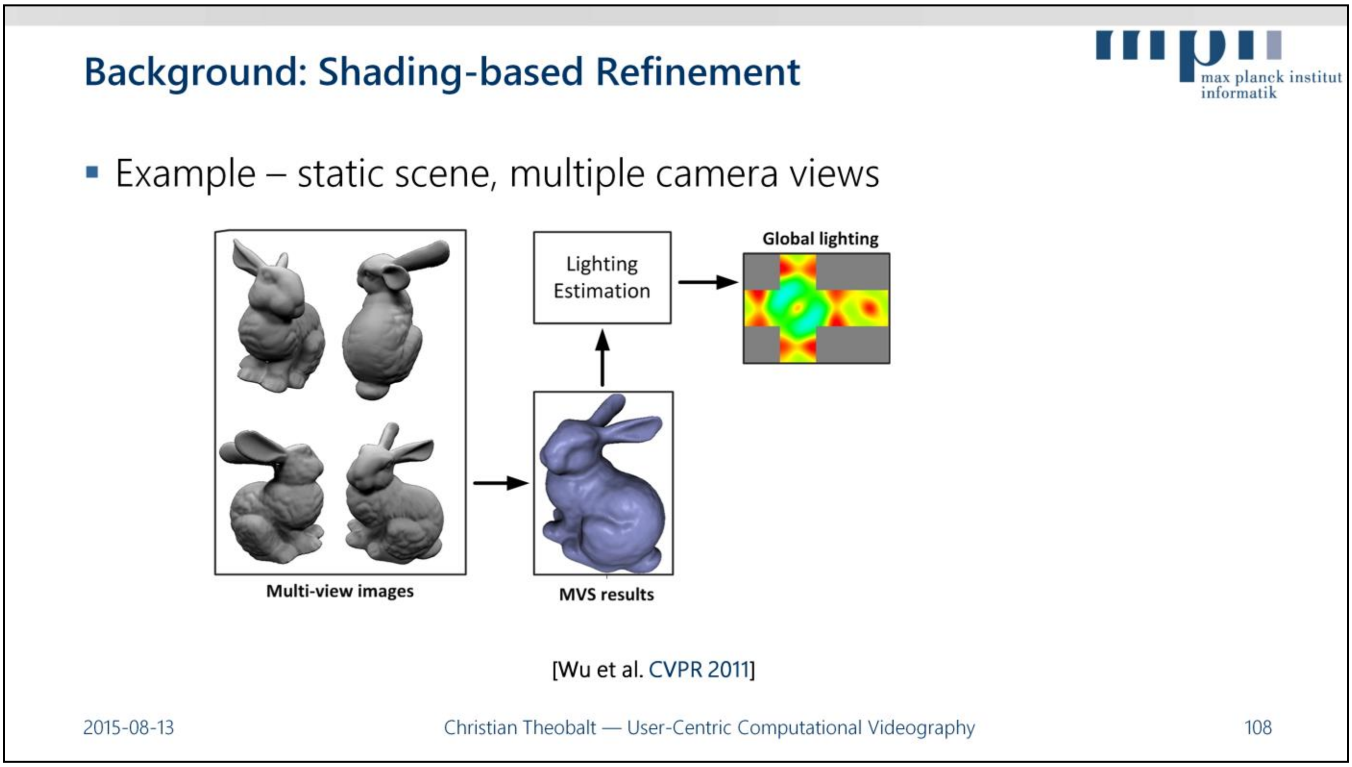
Estimated

Environment map

Spherical harmonics ([Sloan et al. ‘02])

2015-08-13 Christian Theobalt — User-Centric Computational Videography 107

Lighting estimation is formulated as an energy minimization problem. As said before, for a moment it is assumed that the geometry of the object is accurately represented by the coarse initial shape and that the surface has uniform albedo. We now minimize the per-pixel appearance difference between every surface point **x** of the rendered model projected into each view **c** (using camera projection matrix $\mathbf{P_c}$) and the corresponding pixel in the captured image. This yields the energy functional shown above. What we optimize for is an environment map of the incident illumination. To make the problem tractable, we parameterize the spherically parameterized entities on the right hand side of the reflectance equation in the spherical harmonics function basis. More specifically, the incident lighting **L**, as well as the product of **V** and $\max(\omega_i \cdot \mathrm{n}(\mathrm{x}), 0)$ are separately projected into spherical harmonics. Both entities can then be represented with a small number of coefficients. Using this parameterization, the evaluation of the shading equation simplifies to a product of coefficient vectors [Sloan et al. ’02]. What is solved are thus the spherical harmonic coefficients of **L**.



Background: Shading-based Refinement

- Example – static scene, multiple camera views

Multi-view images

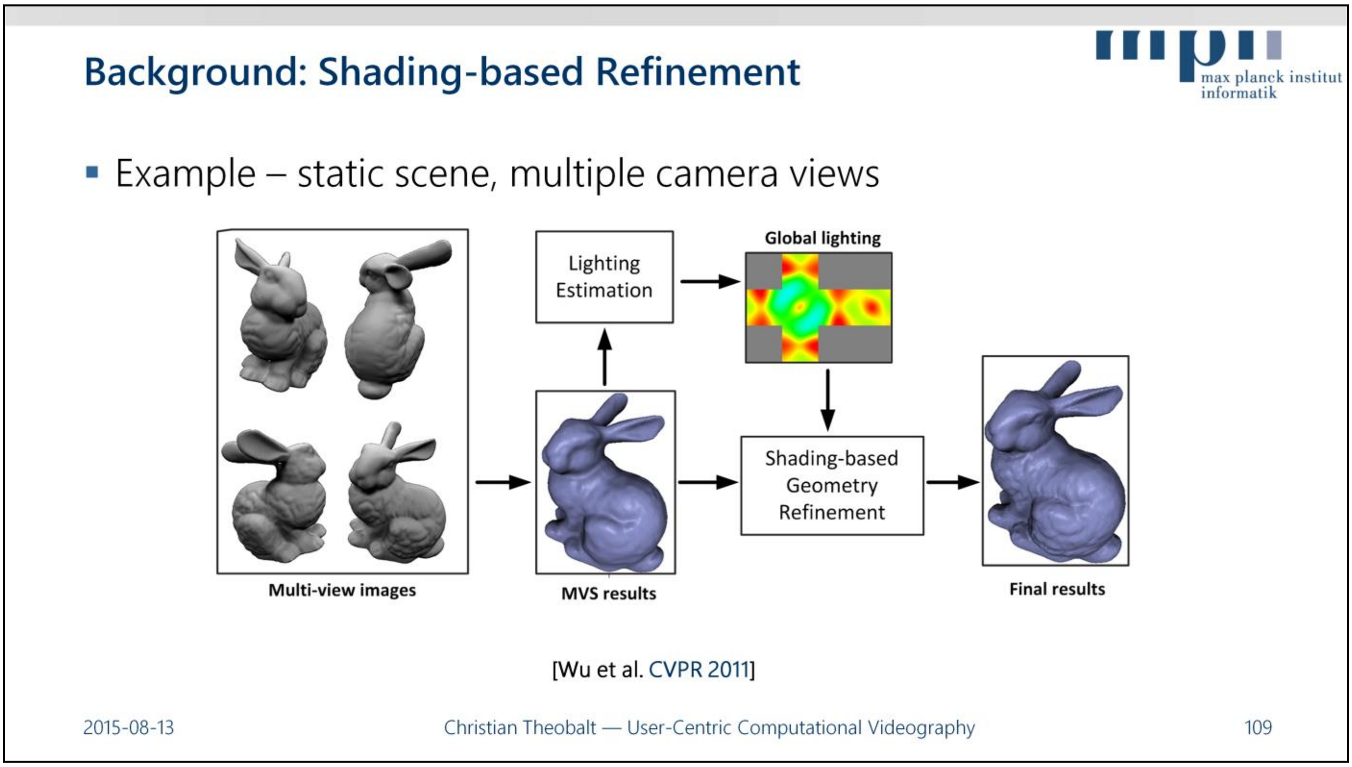
MVS results

Lighting Estimation

Global lighting

[Wu et al. CVPR 2011]

Once the incident lighting is estimated, a second optimization problem is solved to refine the coarse surface geometry using shading cues.



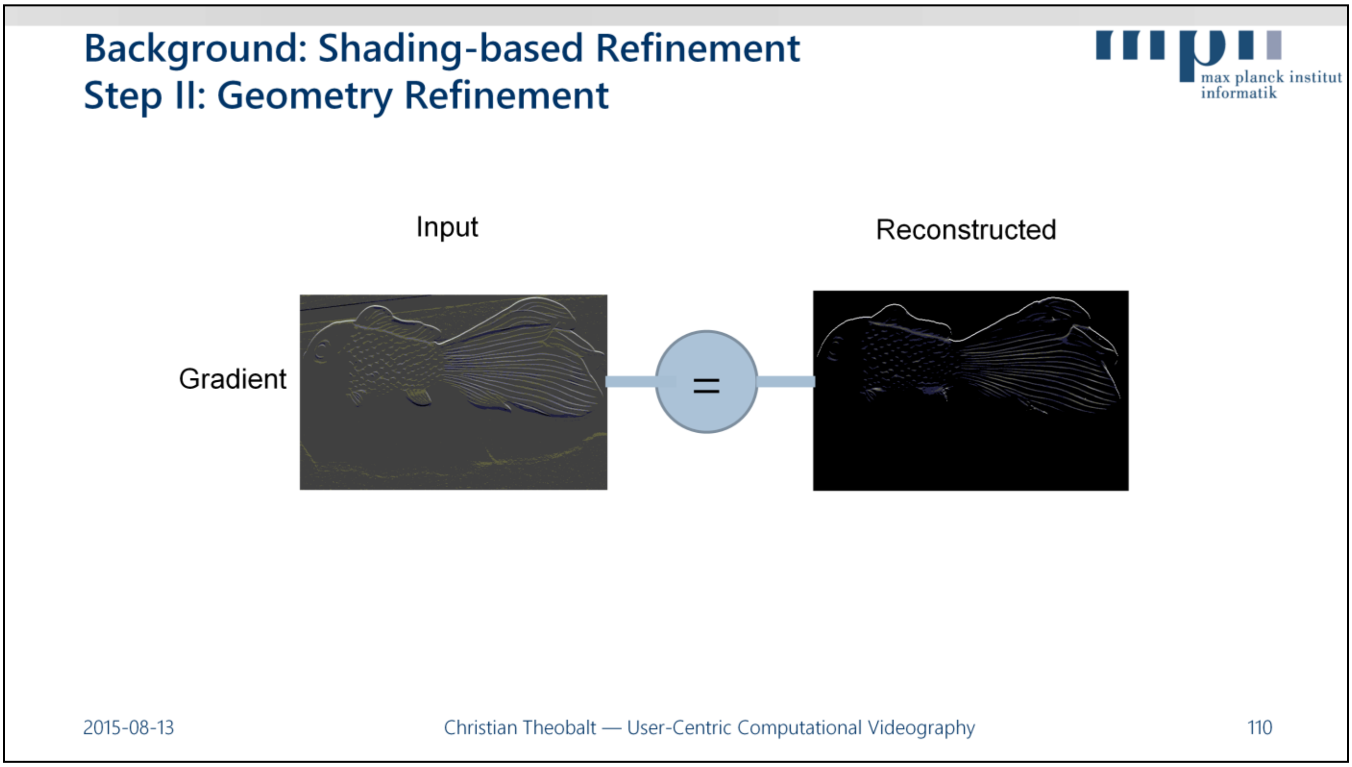
Background: Shading-based Refinement

- Example – static scene, multiple camera views

[Wu et al. CVPR 2011]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 109

Once the incident lighting is estimated, a second optimization problem is solved to refine the coarse surface geometry using shading cues.



Background: Shading-based Refinement
Step II: Geometry Refinement

max planck institut
informatik

Input

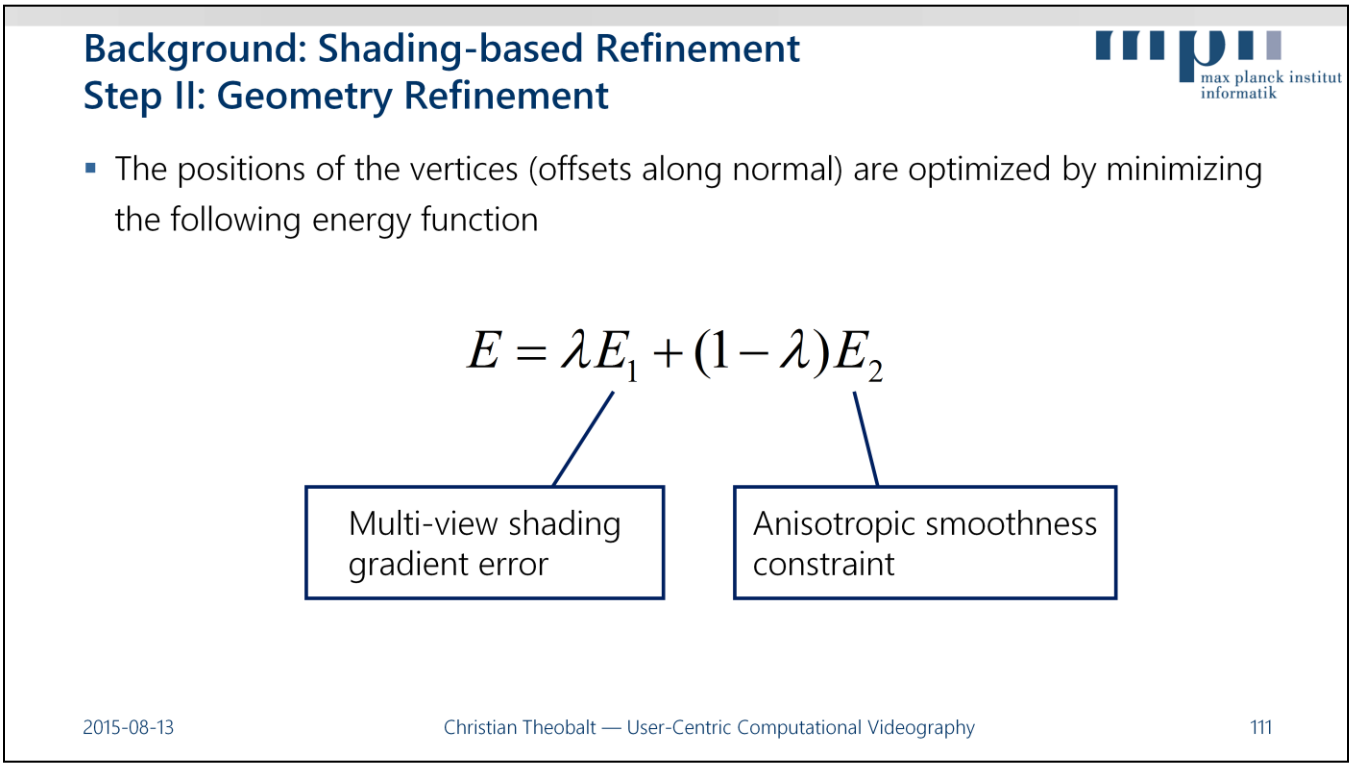
Reconstructed

Gradient

=

Once the incident lighting is estimated, it can be used in a refinement process of the surface geometry. Fine scale shape details often lead to shading gradients on the surface. This effect can be exploited to estimate fine scale surface geometry from images.

Intuitively, the idea is to deform the surface such that the rendered shading gradients match the measured shading gradients in the images, and the deformation of the surface as whole is smooth. This process is again formulated as an optimization problem. The alignment energy penalizes differences in shading gradients rather than absolute intensity values. This makes shape refinement more robust under possible errors in lighting and reflectance estimation.



Background: Shading-based Refinement
Step II: Geometry Refinement

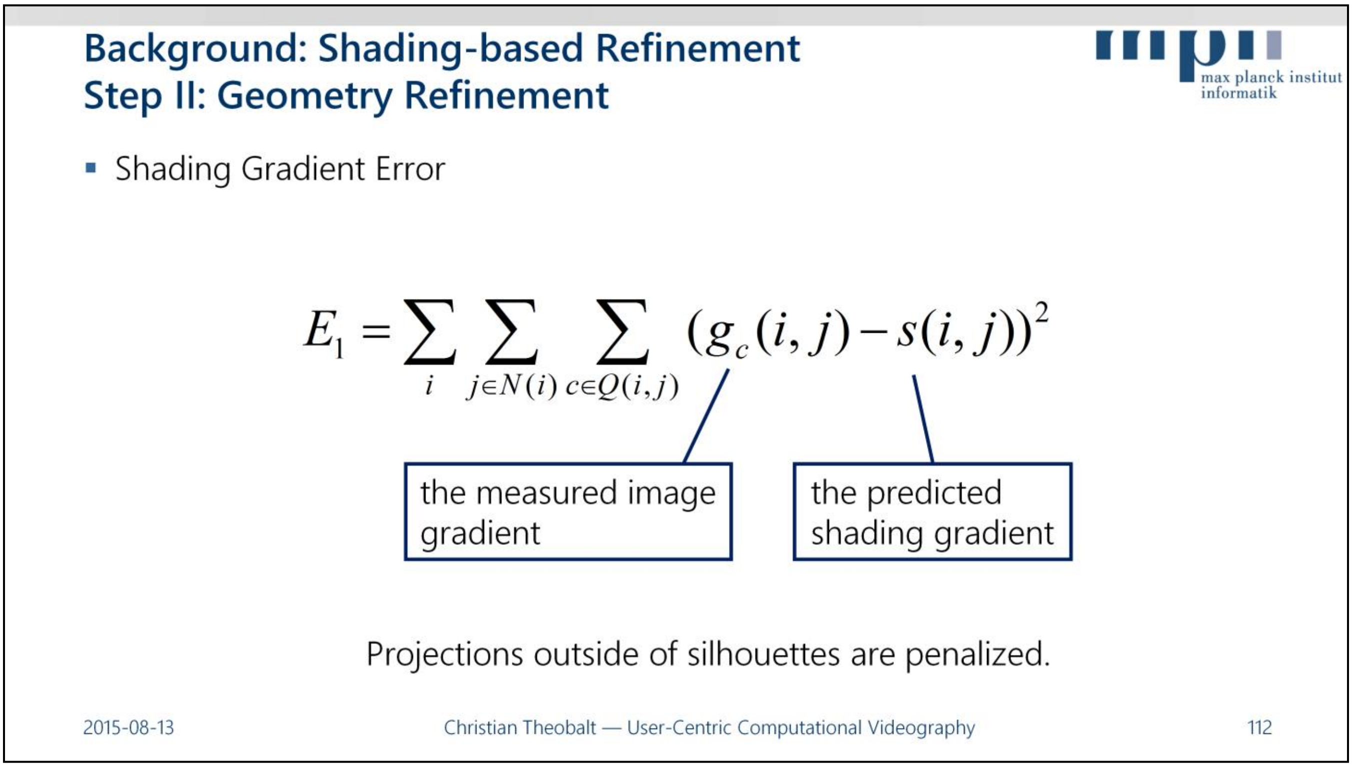
- The positions of the vertices (offsets along normal) are optimized by minimizing the following energy function

$$E = \lambda E_1 + (1 - \lambda) E_2$$

Multi-view shading gradient error

Anisotropic smoothness constraint

The energy optimized for shape refinement consists of a shading gradient error term and an anisotropic smoothing term on the surface.



Background: Shading-based Refinement
Step II: Geometry Refinement

- Shading Gradient Error

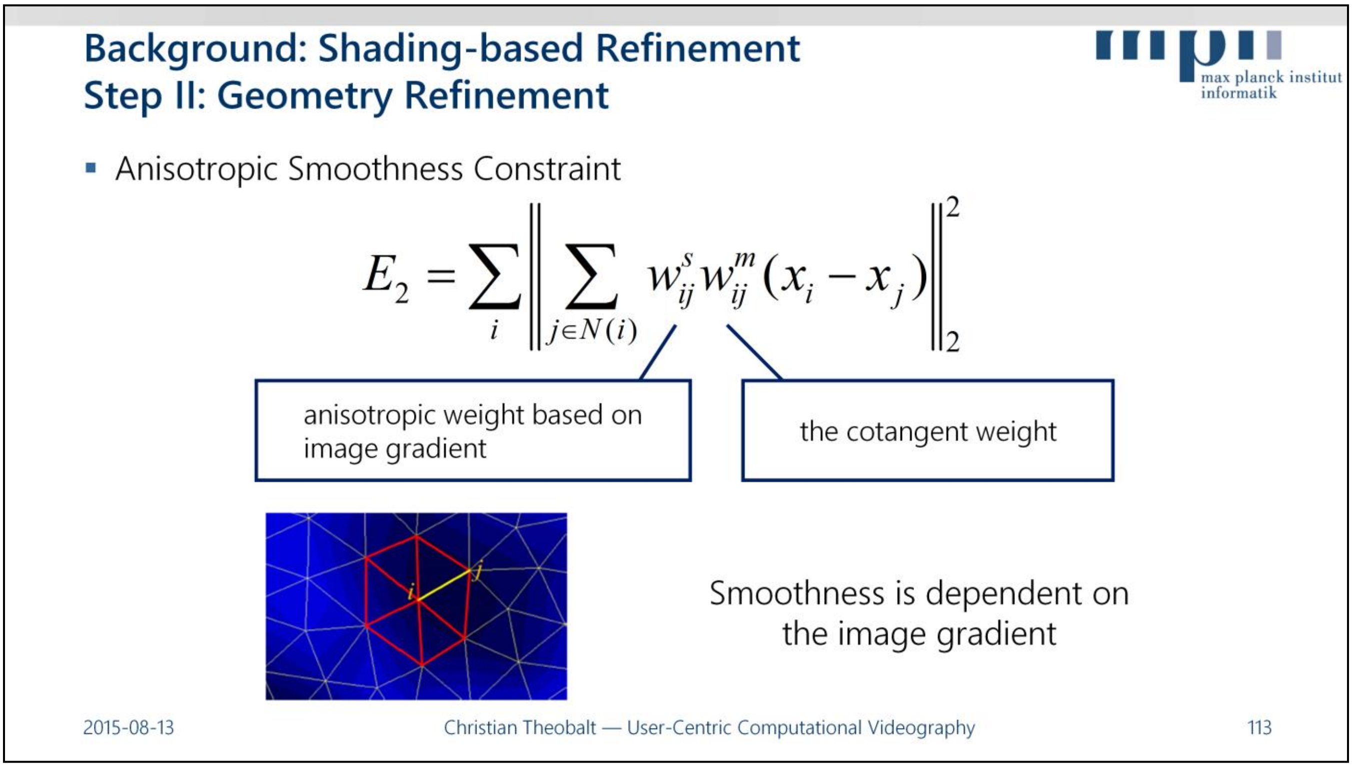
$$E_1 = \sum_{i} \sum_{j \in N(i)} \sum_{c \in Q(i,j)} \left(g_c(i,j) - s(i,j)\right)^2$$

the measured image gradient

the predicted shading gradient

Projections outside of silhouettes are penalized.

The gradient penalization term measures the difference between rendered shading gradients and image gradients. Projections outside of an object’s silhouette in a camera view are penalized.



Background: Shading-based Refinement
Step II: Geometry Refinement

max planck institut
informatik

- Anisotropic Smoothness Constraint

$$E_2 = \sum_i \left\| \sum_{j \in N(i)} w_{ij}^s w_{ij}^m (x_i - x_j) \right\|_2^2$$

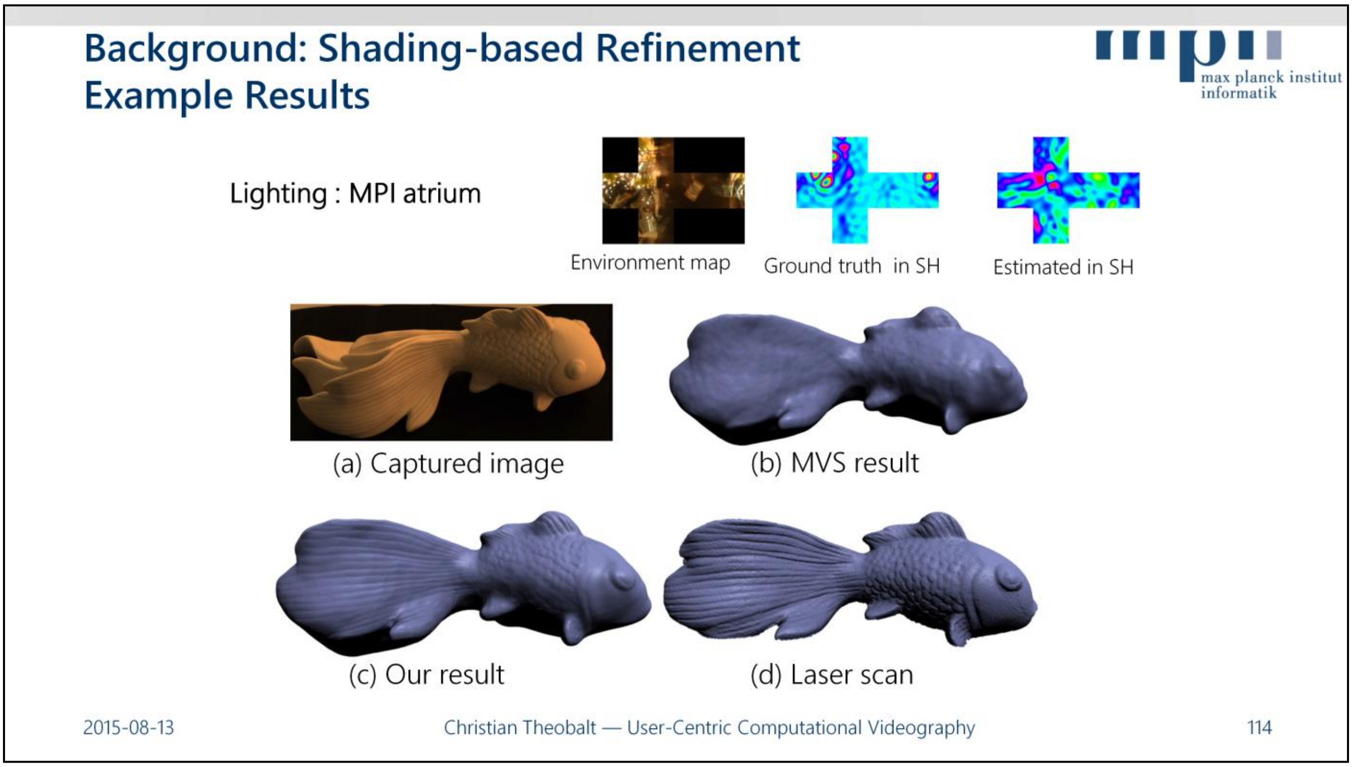
anisotropic weight based on image gradient

the cotangent weight

Smoothness is dependent on the image gradient

2015-08-13 Christian Theobalt — User-Centric Computational Videography 113

The energy comprises an anisotropic smoothness term inspired by mean curvature flow. Essentially the geometry shall be smooth in regions that appear uniformly. In the direction orthogonal to a shading gradient, reconstruction shall also be smooth, in the direction of the gradient geometry shall be less smooth. The first weight above thus depends on the strength of a shading gradient along an edge of the mesh. The second weight corresponds t the cotangent weights used for discretization of the Laplace Beltrami operator, see also [Wu et al. 2011] for more details.



Background: Shading-based Refinement Example Results

Lighting : MPI atrium

Environment map

Ground truth in SH

Estimated in SH

(a) Captured image

(b) MVS result

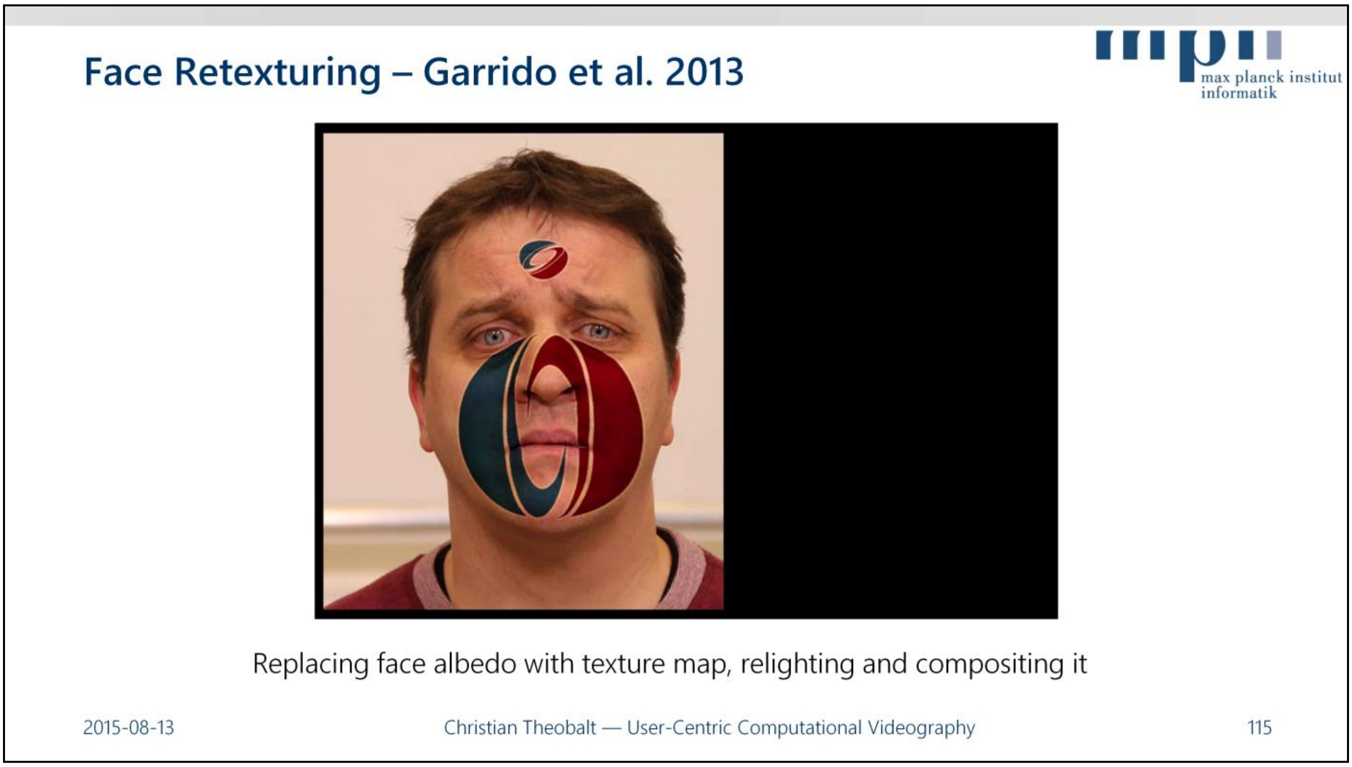
(c) Our result

(d) Laser scan

2015-08-13 Christian Theobalt — User-Centric Computational Videography 114

Here are a few results of a multi-view stereo (MVS) reconstruction of a fish sculpture that was refined with the aforementioned shading-based refinement pipeline (see [Wu et al. 2011]). A comparison against a laser scan shows the high quality of our purely image-based reconstruction. Also the comparison of the reconstructed lighting with the ground truth lighting projected into spherical harmonics demonstrates a faithful estimation.

Please note that for dynamic shape refinement, as used for the face capture results, the above pipeline has been extended to handle spatially-varying albedo, as well as space-time coherent geometry reconstruction [Valgaerts et al. 2012, Garrido et al. 2013].

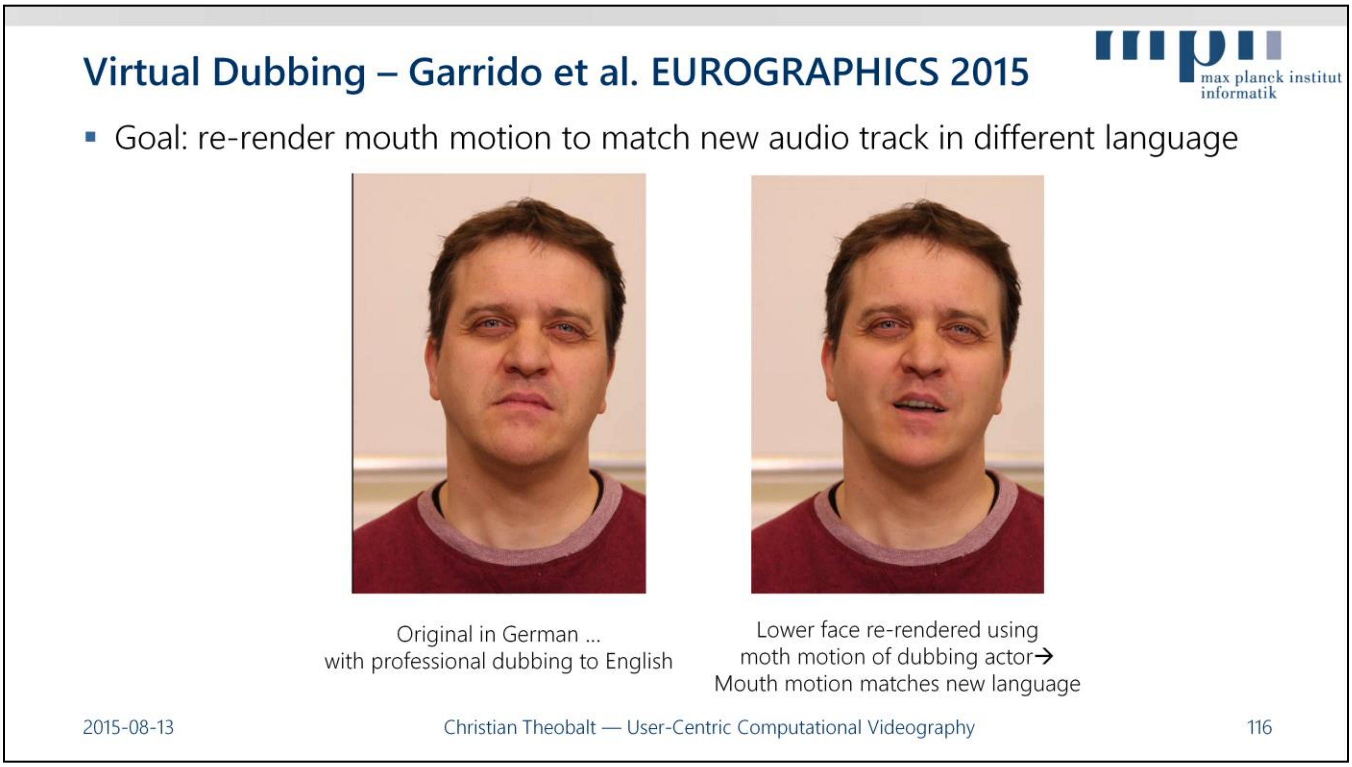


Face Retexturing – Garrido et al. 2013

Replacing face albedo with texture map, relighting and compositing it

2015-08-13 Christian Theobalt — User-Centric Computational Videography 115

After this little detour to explain the background of shape-from-shading-based refinement under general lighting, let’s look at some results on how the approach by Garrido et al. [2013] can be used for model-based video editing. Since albedo, detailed face geometry, and incident illumination are all estimated at high detail, one can edit the albedo map for the face, and re-render it on the captured geometry using the estimated lighting to obtain a photorealistic, temporally stable retexturing composite on the face.



Virtual Dubbing – Garrido et al. EUROGRAPHICS 2015

max planck institut informatik

- Goal: re-render mouth motion to match new audio track in different language

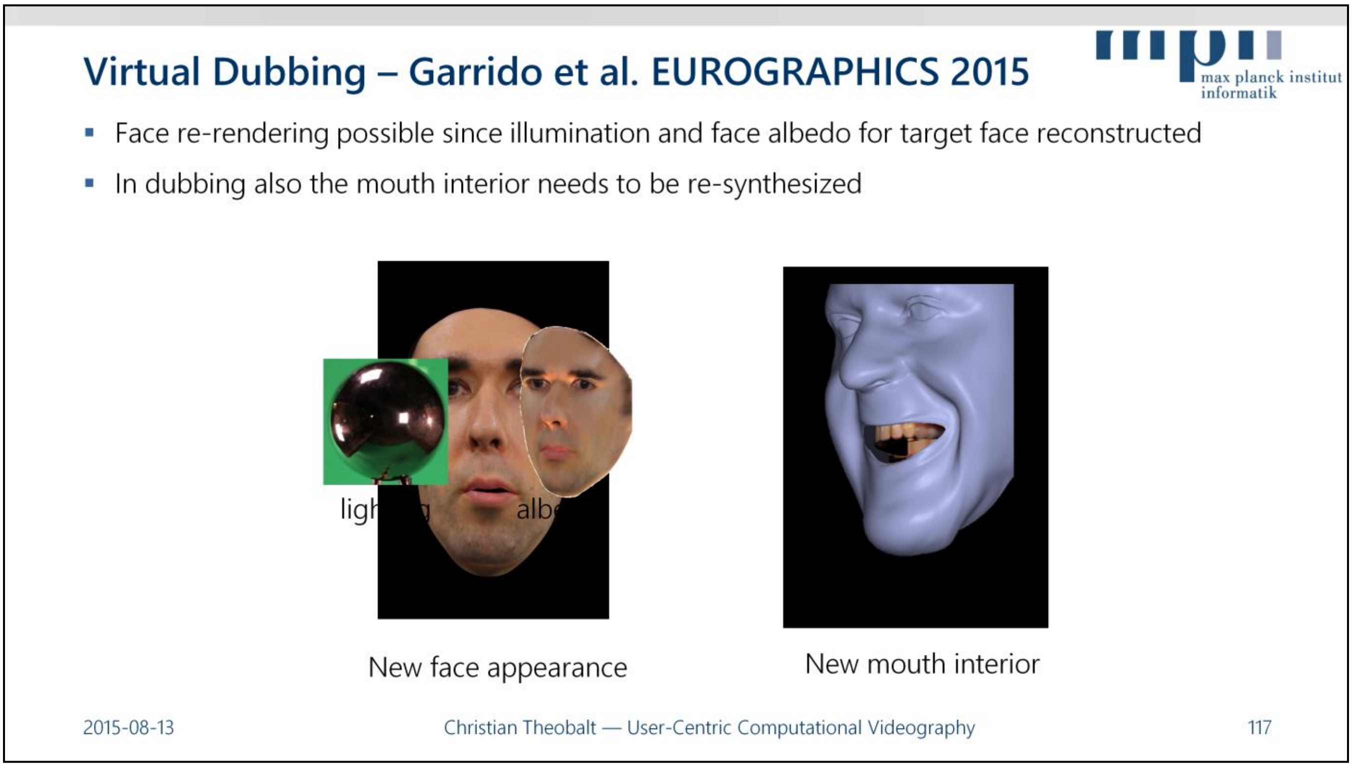
Original in German …
with professional dubbing to English

Lower face re-rendered using
moth motion of dubbing actor→
Mouth motion matches new language

2015-08-13 Christian Theobalt — User-Centric Computational Videography 116

Here is another result from our research that shows how the monocular face performance capture work shown before can be used to achieve advanced video manipulation effects. In many countries it is common practice that movies in a foreign language are dubbed, i.e. a new audio track in the country's language is recorded in a professional dubbing studio and overlaid with the video.

The challenge is to time the new audio such that it still matches the mouth motion of an actor as good as possible. Of course, there can never be a perfect audio-visual alignment, the source and target languages are usually too different. This can easily lead to visual discomfort, and it is the reason why many people dislike dubbed movies. To remedy this problem, we looked into an approach to re-render the lower face motion of an actor in a dubbed video such that it matches the new audio track in the target language [Garrido et al. 2015]. The core idea is to capture both the actor in target video that is to be dubbed, as well as the face motion of the dubbing actor using the monocular face performance capture approach of Garrido et al. 2013. The mouth motion of the dubbing actor can then be used to drive the motion of the target actor's face by means of blend shape coefficient transfer. Since lighting (and in this new work) also a dense face albedo map were estimated, the new mouth motion of the actor can be re-rendered and overlaid with the original video. The mouth interior also needs to be resynthesized, which is done based on a geometric tooth proxy as well as a backdrop for the inner mouth cavity.



Virtual Dubbing – Garrido et al. EUROGRAPHICS 2015

max planck institut informatik

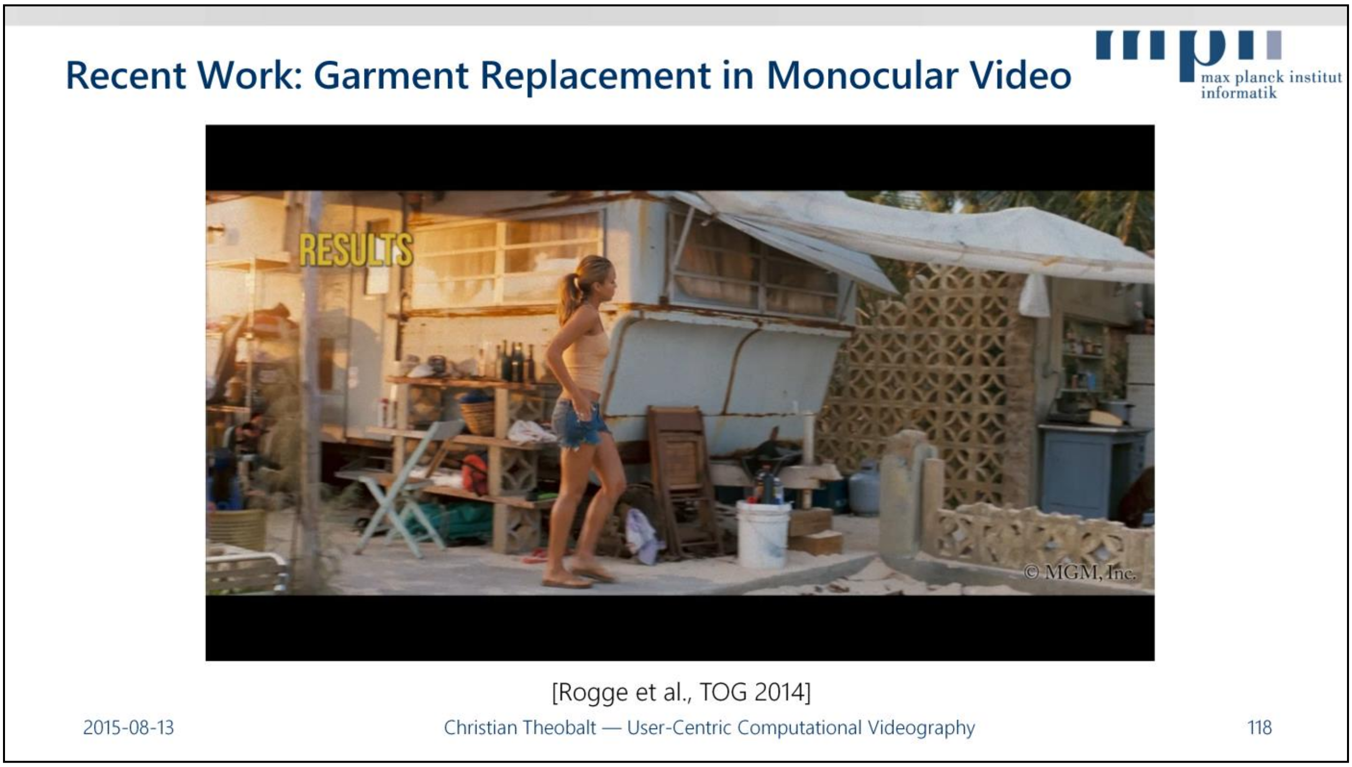
- Face re-rendering possible since illumination and face albedo for target face reconstructed
- In dubbing also the mouth interior needs to be re-synthesized

New face appearance

New mouth interior

2015-08-13 Christian Theobalt — User-Centric Computational Videography 117

As mentioned on the previous slide the appearance and lighting estimation enables us to realistically re-render the target actor’s face in a new pose. Since the mouth motion changes quite notably in the re-renderings for dubbing, the mouth interior needs to also be synthesized. For this, we use a mouth interior proxy with a tooth layer template as shown above.



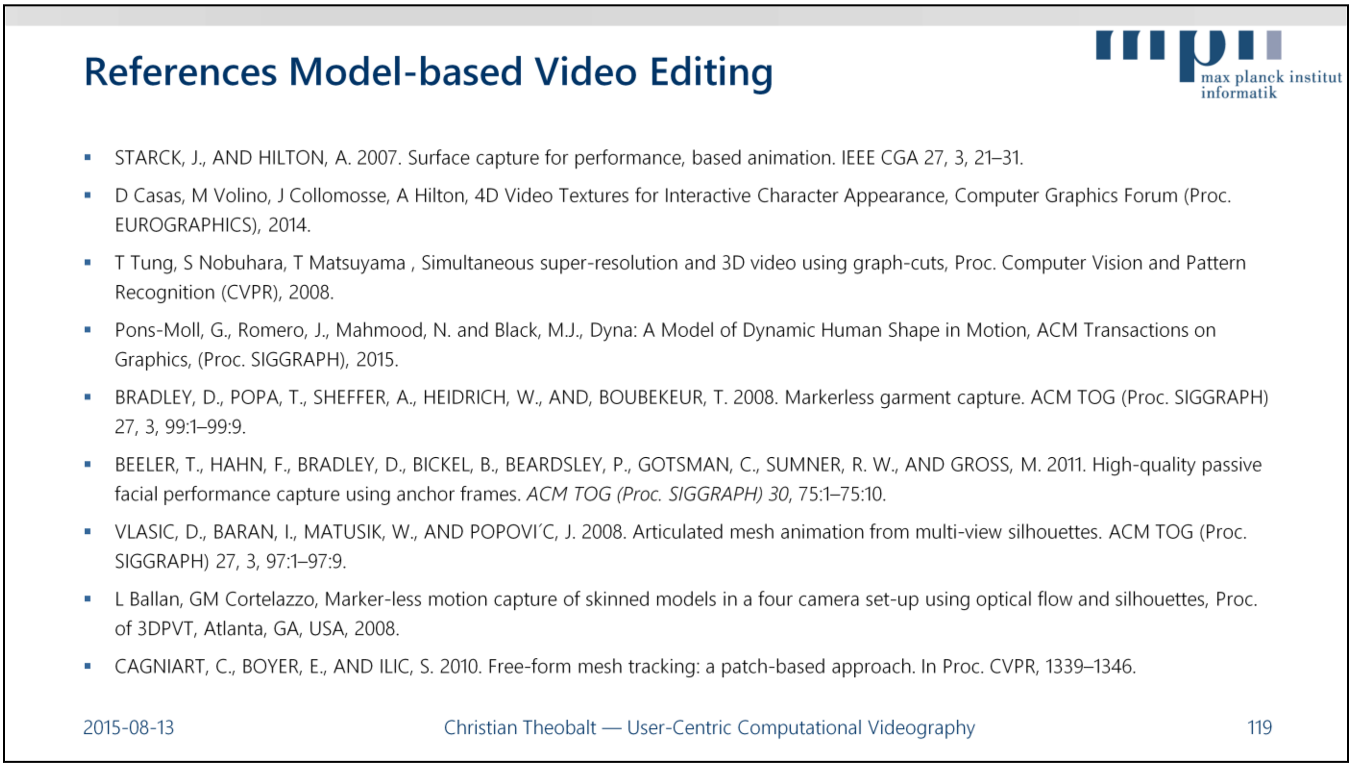
Recent Work: Garment Replacement in Monocular Video

max planck institut informatik

[Rogge et al., TOG 2014]

2015-08-13 Christian Theobalt — User-Centric Computational Videography 118

Let us now take a look at some related model-based video editing approaches proposed recently. The work by Rogge et al. [2014] shows how the clothing of an actor can be replaced in monocular video using a semi-automatic approach. The method also tracks the shape and pose of an actor's body with a parametric human body model, similar to MovieReshape discussed earlier. Using this shape proxy, the scene albedo and incident illumination are coarsely estimated based on a discrete set of spherically arranged light sources as illumination parameterization. Given the moving body geometry and the estimated lighting, moving garment on the body model can be physically simulated and rendered from the camera position of the video using a professional authoring tool. The rendered clothing can then be overlaid with the original video. Since body model reconstruction is not perfect, a feature-based stabilization of the overlay is applied as a post-processing step.



References Model-based Video Editing

- STARCK, J., AND HILTON, A. 2007. Surface capture for performance, based animation. IEEE CGA 27, 3, 21–31.
- D Casas, M Volino, J Collomosse, A Hilton, 4D Video Textures for Interactive Character Appearance, Computer Graphics Forum (Proc. EUROGRAPHICS), 2014.
- T Tung, S Nobuhara, T Matsuyama , Simultaneous super-resolution and 3D video using graph-cuts, Proc. Computer Vision and Pattern Recognition (CVPR), 2008.
- Pons-Moll, G., Romero, J., Mahmood, N. and Black, M.J., Dyna: A Model of Dynamic Human Shape in Motion, ACM Transactions on Graphics, (Proc. SIGGRAPH), 2015.
- BRADLEY, D., POPA, T., SHEFFER, A., HEIDRICH, W., AND, BOUBEKEUR, T. 2008. Markerless garment capture. ACM TOG (Proc. SIGGRAPH) 27, 3, 99:1–99:9.
- BEELER, T., HAHN, F., BRADLEY, D., BICKEL, B., BEARDSLEY, P., GOTSMAN, C., SUMNER, R. W., AND GROSS, M. 2011. High-quality passive facial performance capture using anchor frames. *ACM TOG (Proc. SIGGRAPH) 30*, 75:1–75:10.
- VLASIC, D., BARAN, I., MATUSIK, W., AND POPOVI´C, J. 2008. Articulated mesh animation from multi-view silhouettes. ACM TOG (Proc. SIGGRAPH) 27, 3, 97:1–97:9.
- L Ballan, GM Cortelazzo, Marker-less motion capture of skinned models in a four camera set-up using optical flow and silhouettes, Proc. of 3DPVT, Atlanta, GA, USA, 2008.
- CAGNIART, C., BOYER, E., AND ILIC, S. 2010. Free-form mesh tracking: a patch-based approach. In Proc. CVPR, 1339–1346.

References Model-based Video Editing

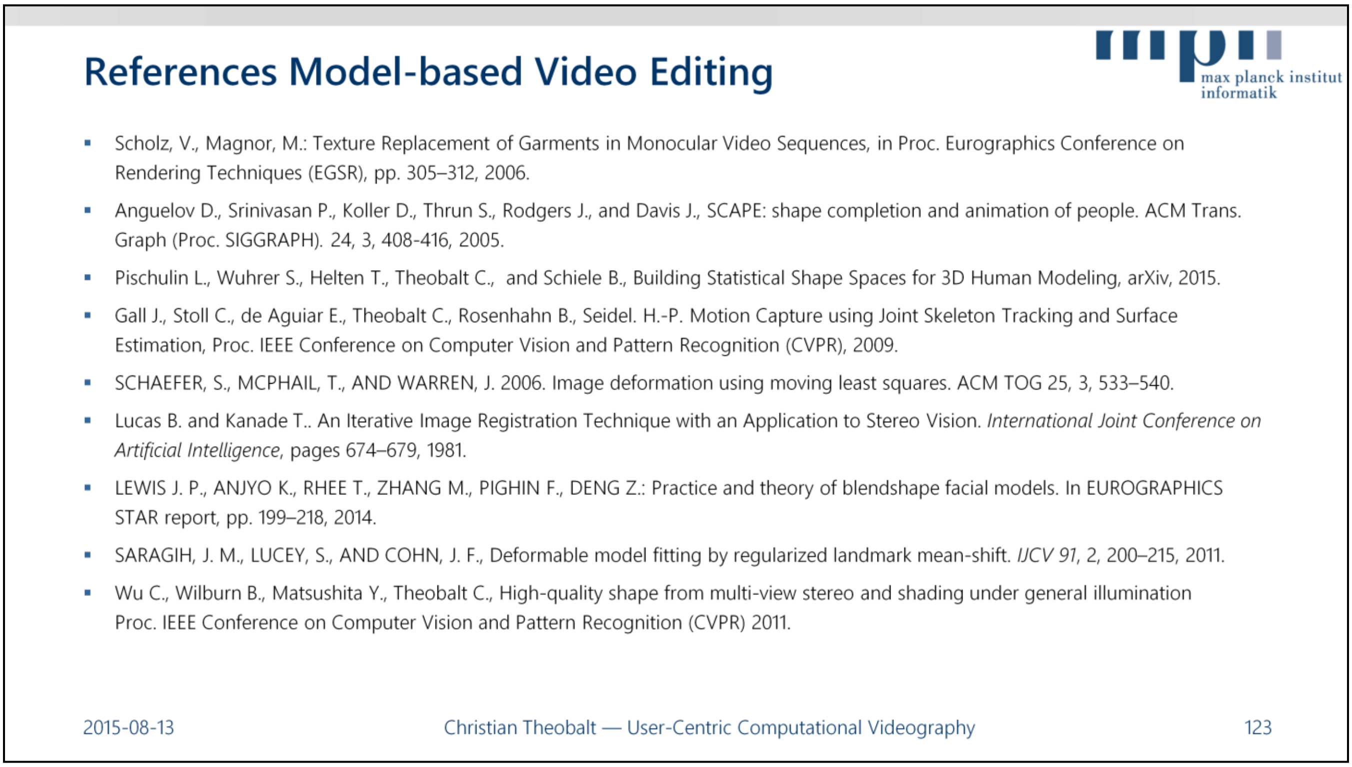
- STARCK, J., AND HILTON, A. 2007. Surface capture for performance, based animation. IEEE CGA 27, 3, 21–31.
- A. Collet, M. Chuang, P. Sweeney, D. Gillett, D. Evseev, D. Calabrese, H. Hoppe, S. Sullivan. High-quality streamable free-viewpoint video. *ACM Transactions on Graphics*, 34(4), 2015.
- VLASIC, D., PEERS, P., BARAN, I., DEBEVEC, P., POPOVIC, J., RUSINKIEWICZ, S., AND MATUSIK, W. 2009. Dynamic shape capture using multi-view photometric stereo. ACM TOG (Proc. SIGGRAPH Asia) 28, 5, 174:1–174:11.
- LIU, Y., STOLL, C., GALL, J., SEIDEL, H.-P., AND THEOBALT, C. 2011. Markerless motion capture of interacting characters using multi-view image segmentation. In Proc. CVPR, 1249–1256.
- DE AGUIAR, E., STOLL, C., THEOBALT, C., AHMED, N., SEIDEL, H.-P., AND THRUN, S. 2008. Performance capture from sparse multi-view video. ACM TOG (Proc. of SIGGRAPH) 27, 98:1–98:10.
- C. Wu, C. Stoll, L. Valgaerts, C. Theobalt,*On-set Performance Capture of Multiple Actors With A Stereo Camera.* In ACM Transactions on Graphics (Proc. of SIGGRAPH Asia) 32, 1-11 (2013).
- C. Theobalt, S. Wuermlin, E. de Aguiar, C. Niederberger, *New Trends in 3D Video* , course at Eurographics 2007, Prague, Czech Republic.
- M. A. Magnor, O. Grau, O. Sorkine-Hornung, C. Theobalt, Digital Representations of the Real World: How to Capture, Model, and Render Visual Reality, ISBN: 9781482243819 - CAT# K23451, Publisher: CRC Press, 2015.
- C. Theobalt, E. de Aguiar, C. Stoll, H-P. Seidel, S. Thrun, Performance Capture from Multi-view Video, in Image and Geometry Procesing for 3D-Cinematography, R. Ronfard and G. Taubin (Eds.), ISBN 978-3-642-12391-7, Springer, 2010.

References Model-based Video Editing

- C. Theobalt, E. de Aguiar, M. Magnor, H.-P. Seidel. Reconstructing human shape, motion and appearance from multi-view video, in Three-dimensional Television , H. Ozaktas, L. Onural (Eds.), ISBN-978-3-540-72531-2, Springer, 2008.
- Valgaerts, L., Wu, C., Bruhn, A., Seidel, H.-P., Theobalt, C.: Lightweight binocular facial performance capture under uncontrolled lighting. ACM Trans. Graph (Proc. SIGGRAPH Asia). 31(6), 187 (2012).
- Wu, C., Stoll, C., Valgaerts, L. and Theobalt: On-set performance capture of multiple actors with a stereo camera.. ACM Trans. Graph. (Proc. SIGGRAPH Asia), 32 (6), Article 161, 2013.
- Shi F., Wu H.T., Tong X., and Chai J.. 2014. Automatic acquisition of high-fidelity facial performances using monocular videos. *ACM Trans. Graph.* 33, 6, 2014.
- C. Cao, D. Bradley, K. Zhou, T. Beeler, Real-Time High-Fidelity Facial Performance Capture, ACM Transactions on Graphics (Proc. SIGGRAPH), 2015.
- Suwajanakorn, S., Kemelmacher-Shlizerman, I., Seitz, S. Total Moving Face Reconstruction, Proc. ECCV 2014.
- Blanz V., Basso C., Poggio T., Vetter T., Reanimating faces in images and video, Computer Graphics Forum, 2003.
- Garrido, P., Valgaerts, L., Wu, C., Theobalt, C.: Reconstructing detailed dynamic face geometry from monocular video. ACM Transactions on Graphics (Proc. SIGGRAPH Asia), 32(6), 2013.
- Ichim, A., Bouaziz, S., Pauly, M., Dynamic 3D Avatar Creation from Hand-held Video Input, ACM Transactions on Graphics (Proceedings of SIGGRAPH), 2015

References Model-based Video Editing

- Cao C., Hou Q., and Zhou K., Displaced dynamic expression regression for real-time facial tracking and animation. ACM Trans. Graph. (Proc. SIGGRAPH), 33, 4, 2014.
- Wu C., Varanasi K., Theobalt C., Full-body performance capture under uncontrolled and varying illumination : A shading-based approach Proc. European Conference on Computer Vision (ECCV), 757 – 770, 2012.
- Wu C., Varanasi K., Liu Y., Seidel H.P., Theobalt C., Shading-based Dynamic Shape Refinement from Multi-view Video under General Illumination, Proc. IEEE International Conference on Computer Vision (ICCV), 1108 – 111, 2011.
- Garrido P., Valgaerts L., Sarmadi H., Steiner I., Varanasi K. , Perez P., Theobalt. C., VDub: Modifying Face Video of Actors for Plausible Visual Alignment to a Dubbed Audio Track, Computer Graphics Forum (Proc. Of EUROGRAPHICS), 2015.
- Richter, M., Varanasi, K., Hasler, N., Theobalt, C., Real-time reshaping of humans, Proc. 3DimPVT, 2012.
- A. Hilsmann and P. Eisert : "Tracking and Retexturing Cloth for Real-Time Virtual Clothing Applications", Mirage 2009 - Computer Vision/Computer Graphics Collaboration Techniques and Applications, Rocquencourt, France, May 2009.
- Salzmann M., Fua P., Linear local models for monocular reconstruction of deformable surfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33 (5), 931-94, 2011.
- Agudo A., Montiel J., Agapito L., Calvo B., Online Dense Non-Rigid 3D Shape and Camera Motion Recovery, Proc. British Machine Vision Conference (BMVC), 2014.
- Vicente S. and Agapito L., Soft inextensibility constraints for template-free non-rigid reconstruction, Proc. European Conference on Computer Vision (ECCV), 2012.

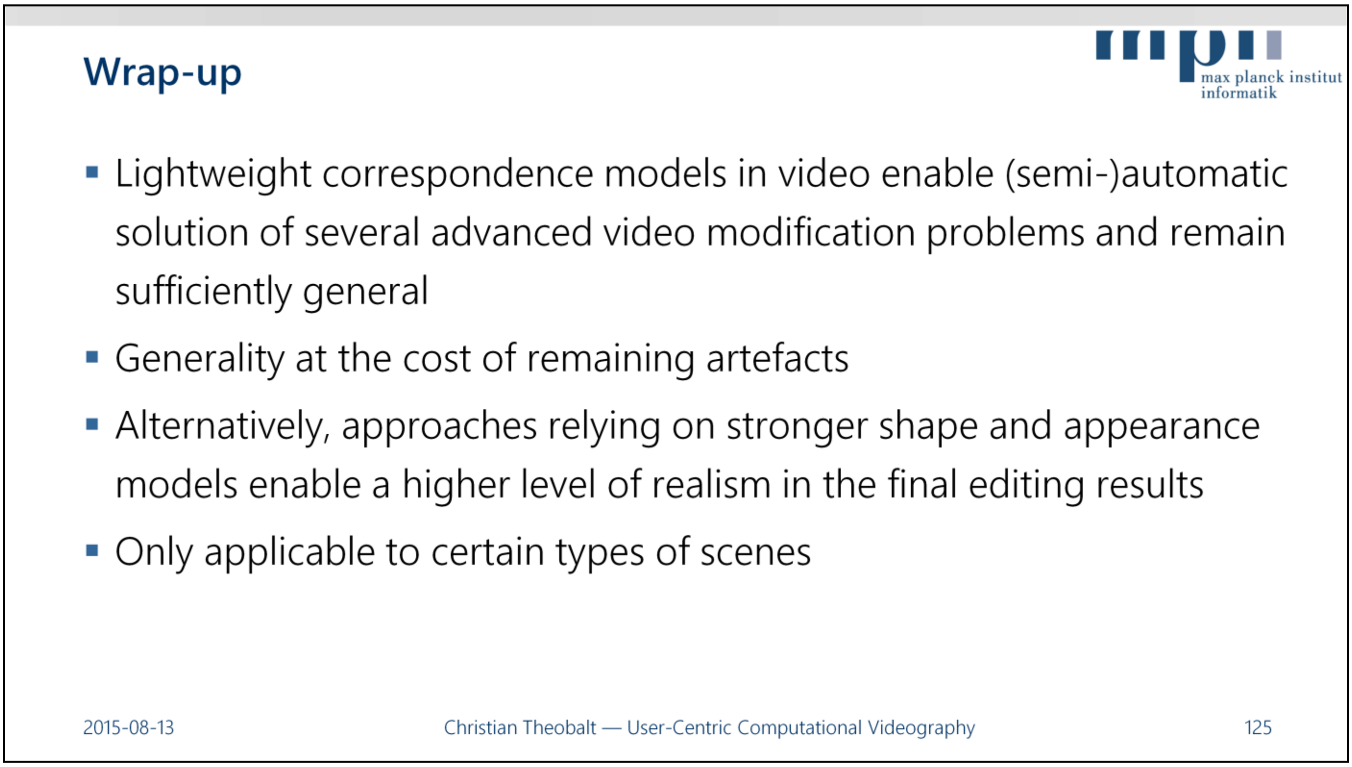


References Model-based Video Editing

- Scholz, V., Magnor, M.: Texture Replacement of Garments in Monocular Video Sequences, in Proc. Eurographics Conference on Rendering Techniques (EGSR), pp. 305–312, 2006.
- Anguelov D., Srinivasan P., Koller D., Thrun S., Rodgers J., and Davis J., SCAPE: shape completion and animation of people. ACM Trans. Graph (Proc. SIGGRAPH). 24, 3, 408-416, 2005.
- Pischulin L., Wuhrer S., Helten T., Theobalt C., and Schiele B., Building Statistical Shape Spaces for 3D Human Modeling, arXiv, 2015.
- Gall J., Stoll C., de Aguiar E., Theobalt C., Rosenhahn B., Seidel. H.-P. Motion Capture using Joint Skeleton Tracking and Surface Estimation, Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.
- SCHAEFER, S., MCPHAIL, T., AND WARREN, J. 2006. Image deformation using moving least squares. ACM TOG 25, 3, 533–540.
- Lucas B. and Kanade T.. An Iterative Image Registration Technique with an Application to Stereo Vision. *International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.
- LEWIS J. P., ANJYO K., RHEE T., ZHANG M., PIGHIN F., DENG Z.: Practice and theory of blendshape facial models. In EUROGRAPHICS STAR report, pp. 199–218, 2014.
- SARAGIH, J. M., LUCEY, S., AND COHN, J. F., Deformable model fitting by regularized landmark mean-shift. *IJCV 91*, 2, 200–215, 2011.
- Wu C., Wilburn B., Matsushita Y., Theobalt C., High-quality shape from multi-view stereo and shading under general illumination Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011.

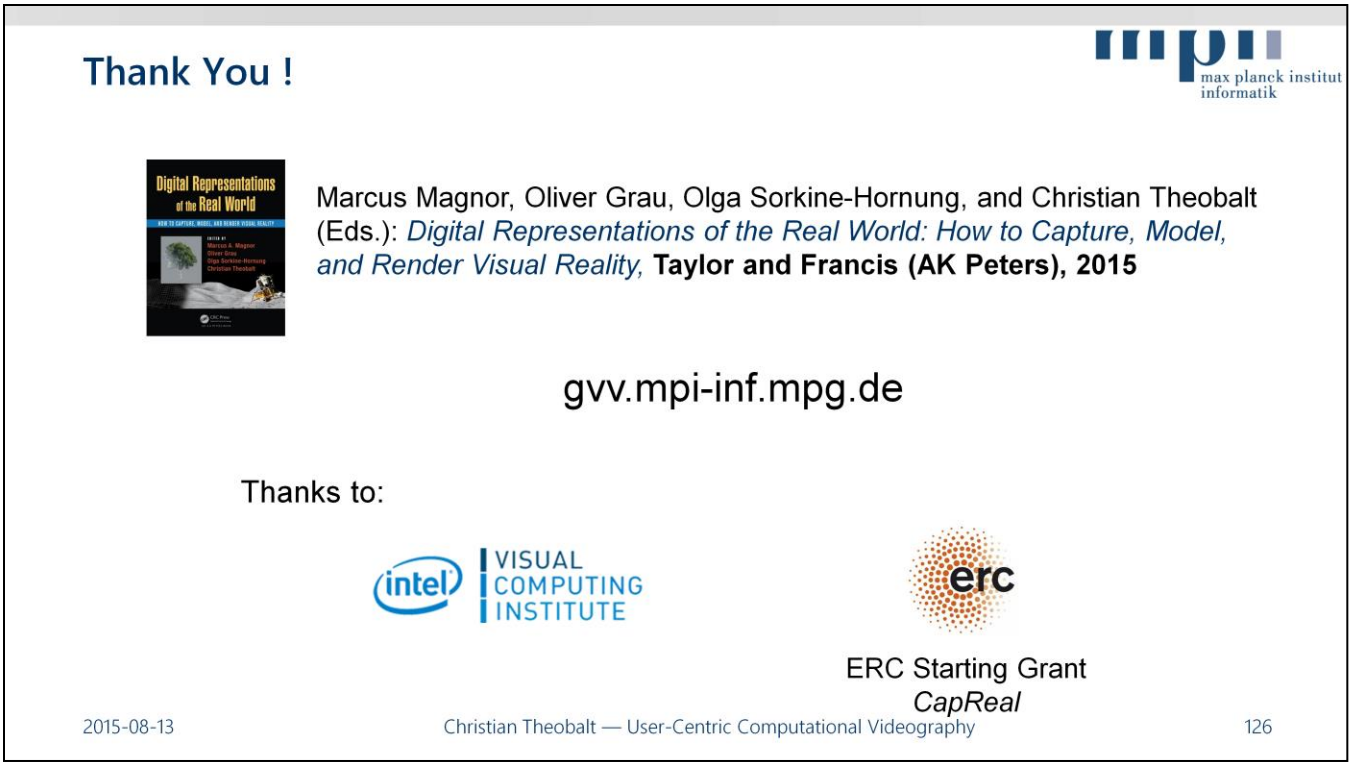
References Model-based Video Editing

- P.-P. Sloan, J. Kautz, and J. Snyder. Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. In ACM TOG (Proc. SIGGRAPH), pages 527–536, New York, NY, USA, 2002.
- Garrido P., Valgaerts L., Sarmadi H., Steiner I., Varanasi K., Perez P., and Theobalt C., "vDub: Modifying face video of actors for plausible visual alignment to a dubbed audio track.", Computer Graphics Forum (Proc. EUROGRAPHICS), 2015.
- Rogge L., Klose F., Stengel M., Eisemann M., and Magnor M., Garment Replacement in Monocular Video Sequences, ACM Transactions on Graphics, vol. 34, no. 1, pp. 6:1–6:10, 2014.



Wrap-up

- Lightweight correspondence models in video enable (semi-)automatic solution of several advanced video modification problems and remain sufficiently general
- Generality at the cost of remaining artefacts
- Alternatively, approaches relying on stronger shape and appearance models enable a higher level of realism in the final editing results
- Only applicable to certain types of scenes



Thank You !

max planck institut
informatik

Marcus Magnor, Oliver Grau, Olga Sorkine-Hornung, and Christian Theobalt (Eds.): *Digital Representations of the Real World: How to Capture, Model, and Render Visual Reality,* **Taylor and Francis (AK Peters), 2015**

gvv.mpi-inf.mpg.de

Thanks to:

intel
VISUAL
COMPUTING
INSTITUTE
erc

ERC Starting Grant
CapReal