

Layered Depth Images for Multi-View Coding

Vincent Jantet

ENS-Cachan, Antenne de Bretagne, Campus de Ker Lann, 35170 Bruz – France

INRIA Rennes, Bretagne Atlantique, Campus de Beaulieu, 35042 Rennes – France

Ph.D. Thesis defense, Rennes, 2012

Collaboration with IRISA, INSA and Brittany Region in Futurim@ge project



Applicative context

Functionalities

3DTV: Depth feeling by stereo-vision simulation

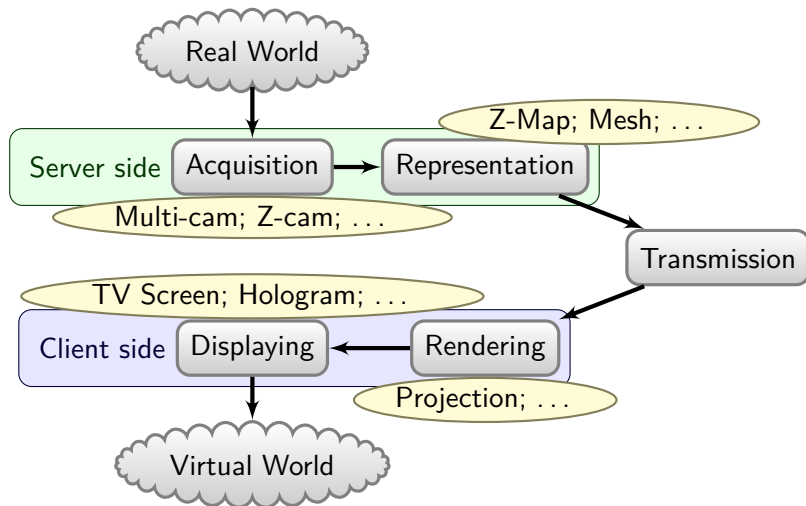
FVV: Live viewpoint selection



3DTV

FVV

3D video processing scheme



Each choice has an impact on following steps

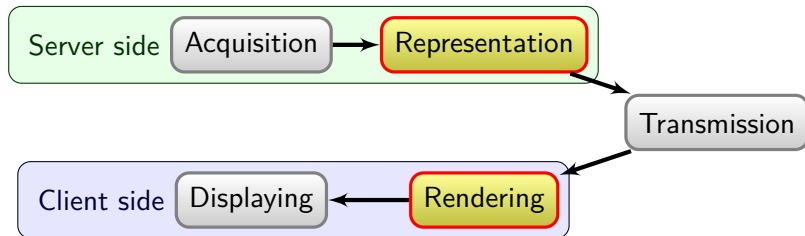
Thesis objectives

Intermediate representation

- Compact
- Bit-Rate scalable

Virtual View rendering

- Fast
- Accurate



SoA: Rendering-optimized representations

- Multi-View Videos
- Plenoptic Function (Light Ray)
- Microfacet Billboarding
- ...

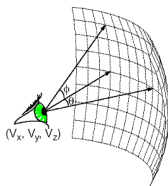
[DTM96]

[AB91]

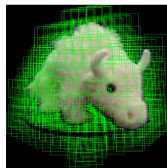
[YSK⁺02]



Multi-View Video



Plenoptic Function



Microfacet Billboarding

Advantages

Photo-realistic rendering

Limitations

Huge amount of data

SoA: Transmission-optimized representations

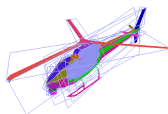
- 2D plus depth video (2D+Z) [ISO07]
- Layered Depth Image (LDI) [SGHS98]
- Billboard Cloud [DDSD03]
- Polygon Mesh
- ...



2D+Z



LDI



Billboard Cloud



Polygon Mesh

Advantages

Compact representation

Limitations

Hard to construct from real scene

Contributions

LDI representation

Compact representation
(naturally remove correlations)

JPF rendering method

Point-based projection
method which handle artifacts

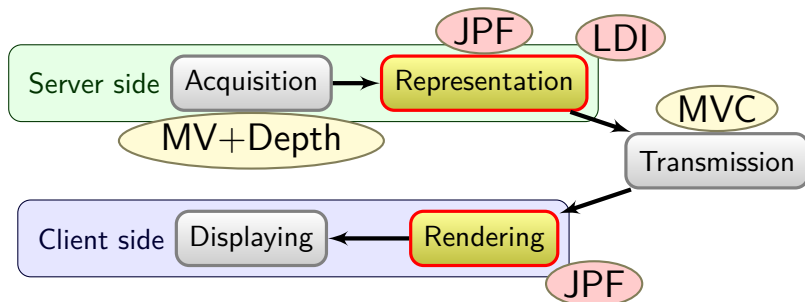


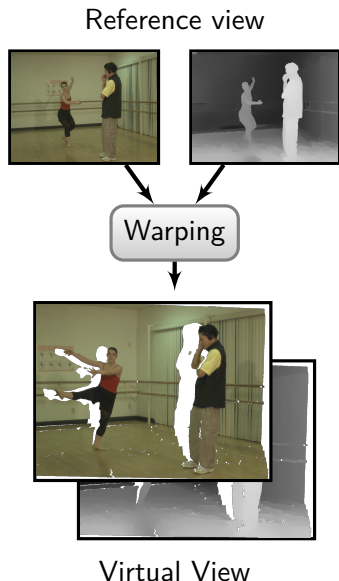
Table of contents

- 1 View synthesis (JPF)
- 2 Layered Depth Image (LDI)
- 3 LDI-based multi-view compression
- 4 Conclusions

Table of contents

- 1 View synthesis (JPF)
 - Projection algorithm
 - Joint Projection Filling (JPF)
 - Rendering results
- 2 Layered Depth Image (LDI)
- 3 LDI-based multi-view compression
- 4 Conclusions

View synthesis: Classical Warping algorithm



View synthesis methods use projection algorithm (warping)

Warping algorithm

Geometrical projection

Input: Texture + Depth map

From reference View Point
+ Cameras parameters

Output: Texture + Depth map

Seen from new View Point

View Synthesis: Warping common artifacts

Disocclusions: Occluded areas which become visible

Cracks: Small holes due to sampling

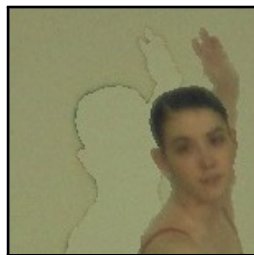
Ghosting: Boundaries pixels with mixed foreground/background color



Disocclusions

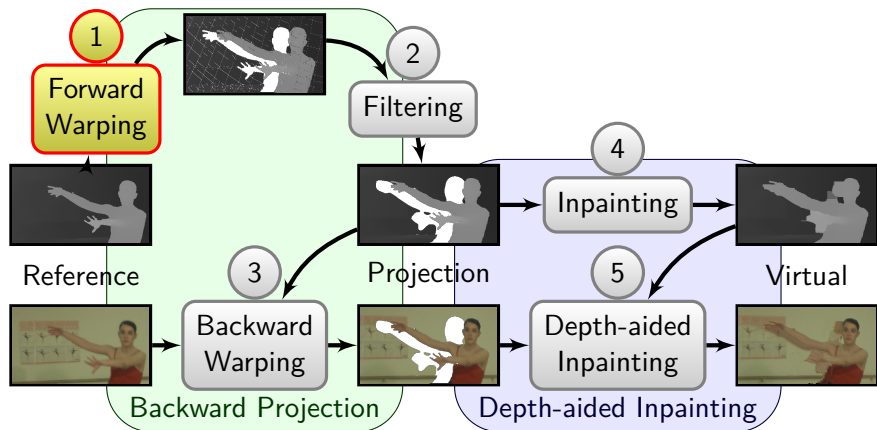


Cracks



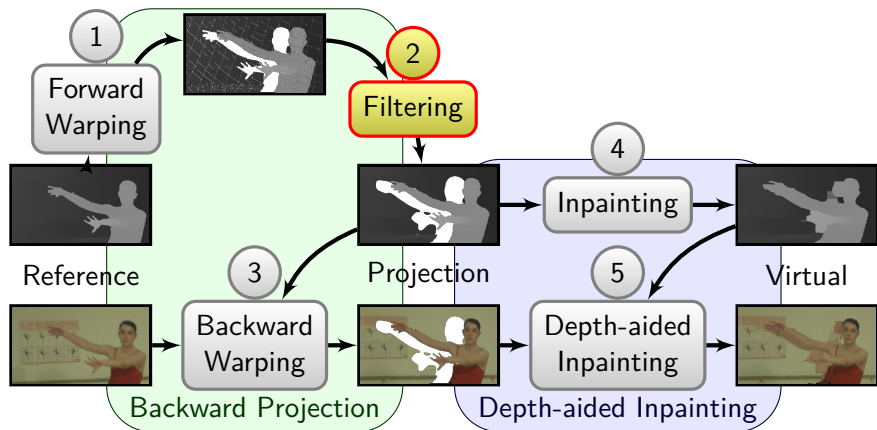
Ghosting

View Synthesis: Classical scheme



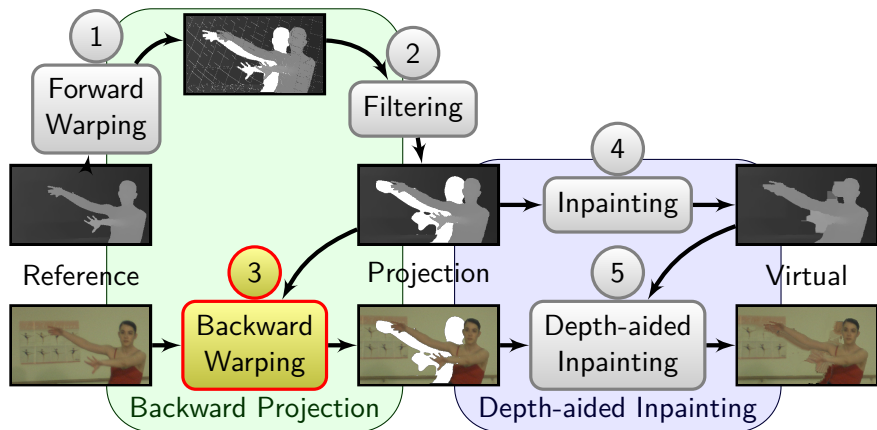
1 Forward Warping: Lose pixels connectivity

View Synthesis: Classical scheme



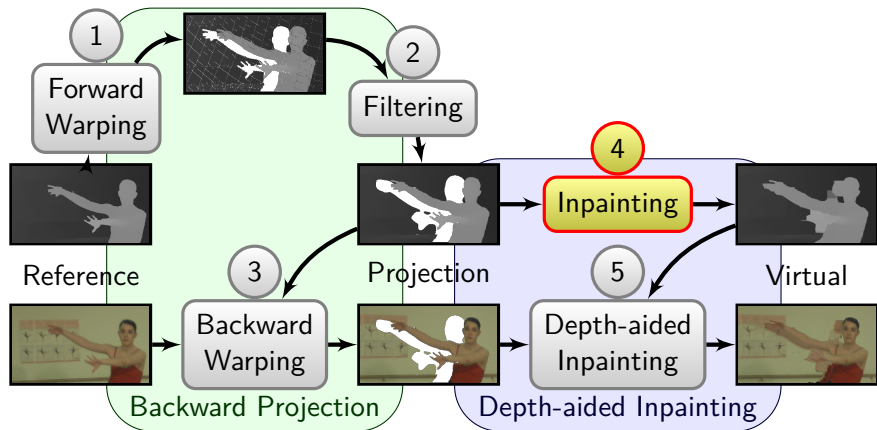
2 Filtering: Fills Cracks and avoids Ghosting

View Synthesis: Classical scheme



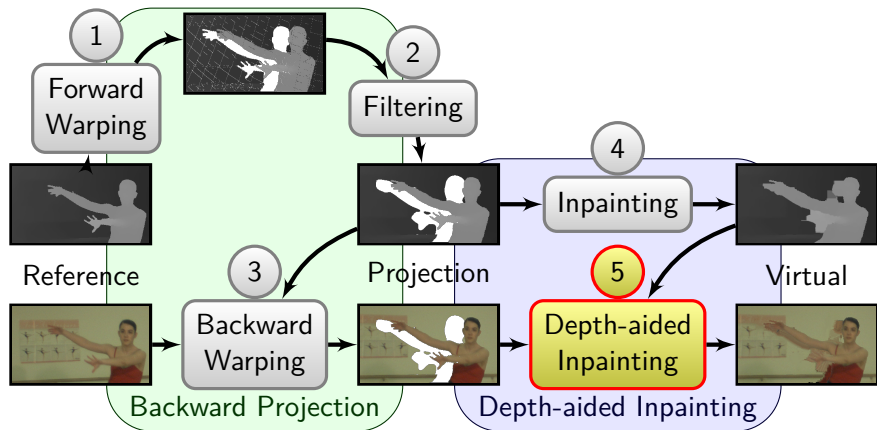
3 Backward Warping: Retrieves color from reference view

View Synthesis: Classical scheme



4 Depth Inpainting: Fills disocclusions with mixed FG/BG depth

View Synthesis: Classical scheme



5 Depth-aided Inpainting: Fills disocclusions with mixed FG/BG texture

View Synthesis: Classical scheme

Limitations

Forward Warping: Lose connectivity

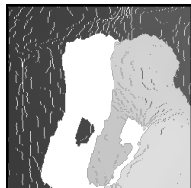
Depth Inpainting: Can not retrieve structure

Texture Inpainting: May fill BG with FB texture

Errors are amplified along the process

Need for an accurate virtual depth
map synthesizing method

Introducing a new Joint Projection Filling method



Forward Proj.



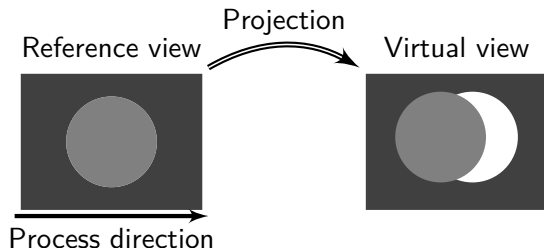
Dir. inpaint.

McMillan [McM95]

Pixel scanning order to avoid
the use of a zBuffer

Contribution

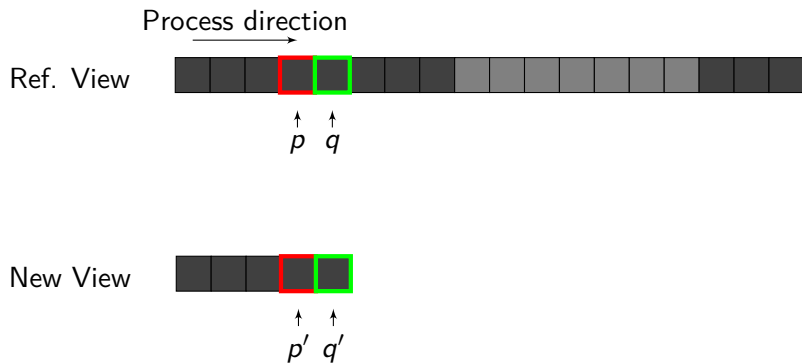
Also provides pixels
connectivity information



Projection without
zBuffer

BackGround pixels are projected before ForeGround pixels

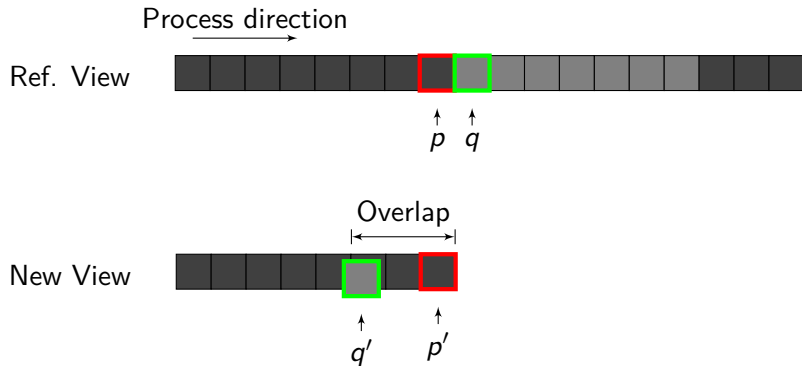
JPF: For rectified views



Consider p and q two pixels projected on p' and q'

$$\left\{ \begin{array}{l} q'_x = p'_x + 1 \quad \text{No artifact} \end{array} \right.$$

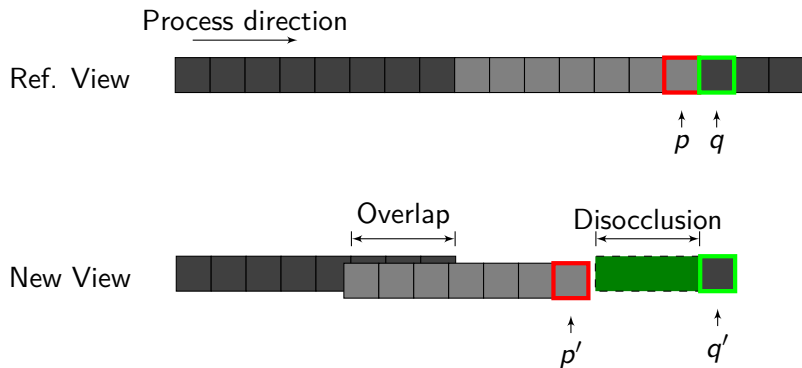
JPF: For rectified views



Consider p and q two pixels
projected on p' and q'

$$\begin{cases} q'_x = p'_x + 1 & \text{No artifact} \\ q'_x < p'_x + 1 & \text{Overlap} \end{cases}$$

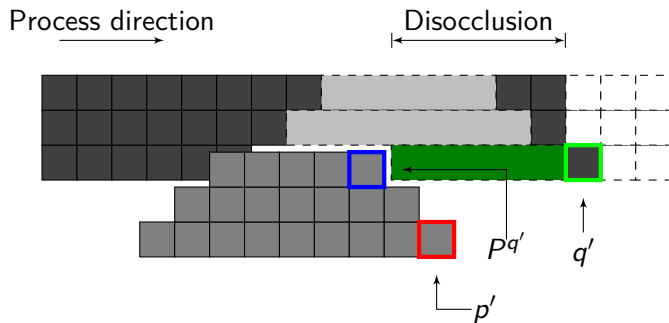
JPF: For rectified views



Consider p and q two pixels projected on p' and q'

$$\begin{cases} q'_x = p'_x + 1 & \text{No artifact} \\ q'_x < p'_x + 1 & \text{Overlap} \\ q'_x > p'_x + 1 & \text{Disocclusion} \end{cases}$$

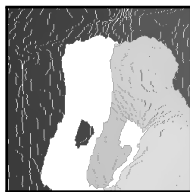
JPF Generalized: For non rectified views



$P^{q'}$: the last pixel projected
on row q'

$$\begin{cases} q'_x \leq P_x^{q'} + 1 & \text{No artifact} \\ q'_x > P_x^{q'} + 1 & \text{Disocclusion} \end{cases}$$

Process direction
←



Forward Proj.

Navier-Stokes

Directional inpaint.

JPF Proj.

JPF method well synthesizes sharp boundaries and thin fingers

Advantages

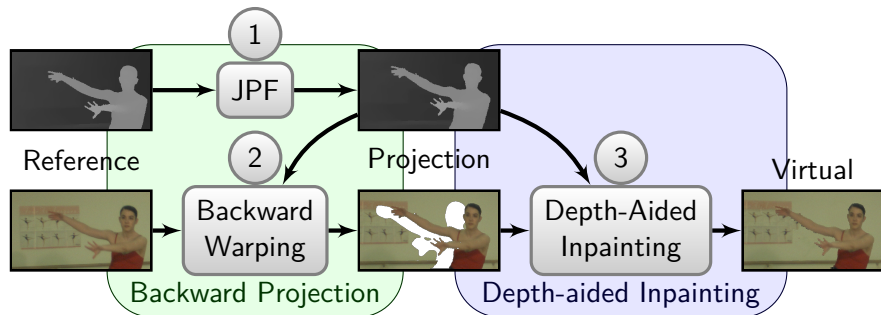
- One-step projection, without post-processing
- Handles cracks and disocclusions during the projection
- Fills disocclusions with background
- Preserves geometrical structures

Limitations

- Hard to implement on GPU
- Introduce stretching artifacts if used for texture projection

Should be used as a part of a full view synthesis method

View Synthesis: Proposed scheme



JPF method replaces for:

- Forward Projection
- Depth Filtering
- Depth Inpainting

Synthesized Depth used for:

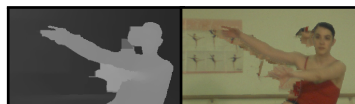
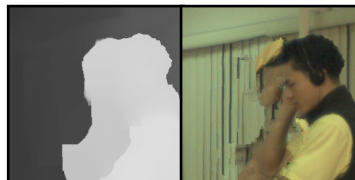
- Backward Warping
- Depth-Aided Inpainting

View Synthesis rendering results

Disocclusions



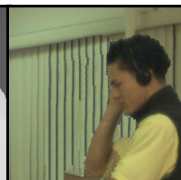
SoA inpaint.¹ Daribo's DAI²



JPF proj.



Full-Z DAI³



Inconsistent virtual depth map \Rightarrow Texture artifacts
JPF synthesize correct depth map which helps DAI

¹Navier-Stokes's inpainting [BBS01]

²Daribo's Depth Aided Inpainting [DP10]

³Full-Z Depth Aided Inpainting [Jantet et al., 2011a] (inspired from Daribo)

Conclusions on Virtual view synthesis

From a single input video + depth:

JPF: Synthesize virtual depth map

DAI: Recreate missing texture

Realistic synthesized video, but:

- Introduces temporal flickering
- Incoherence between two synthesized views

From multi-view + depth:

Could retrieve **real** disocclusions textures \Rightarrow Introducing LDI

Table of contents

- 1 View synthesis (JPF)
- 2 Layered Depth Image (LDI)
 - Definition
 - Classical LDI construction
 - Incremental-LDI construction (I-LDI)
 - Object-based classification (O-LDI)
- 3 LDI-based multi-view compression
- 4 Conclusions

LDI: Layered Depth Image

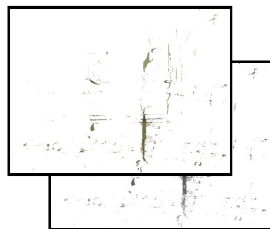
Set of pixels, from a reference viewpoint, organized in layers



1st layer (visibles pixels)



2nd layer (occluded pixels)



3rd layer (...)

...

Advantages

Disocclusion: Could be filled by **real** texture

Camera freedom: Virtual camera can move inside a large area

Compactness: Eliminate some correlated pixels and reduce data size

Every input views are warped
onto a reference viewpoint,
and then merged together

Merging policy

Eliminates duplicated
pixels

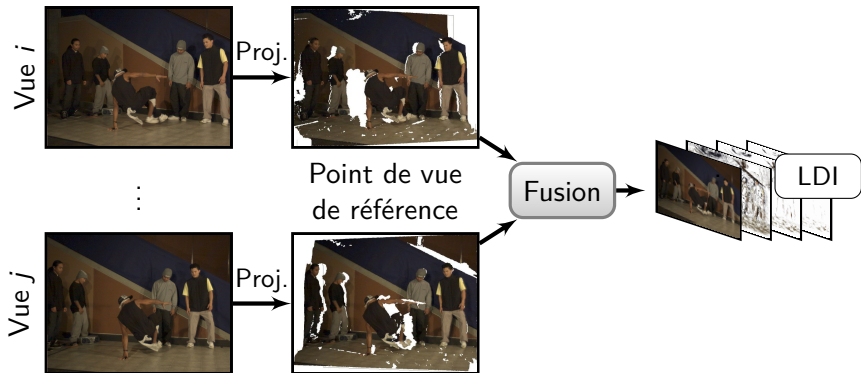


Schéma de construction naïve des LDI.

Classical LDI limitations

Redundancies

- Many pixels in many layers, partially empty
- Scattered pixels distribution

Introducing Incremental LDI construction

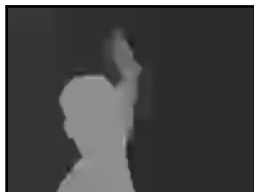


Scattered distribution

Compression artifacts

- Large depth discontinuities
- Motion in multi-layer
- Boundaries in multi-layer
- Uneasily compressed

Introducing Object-based LDI representation

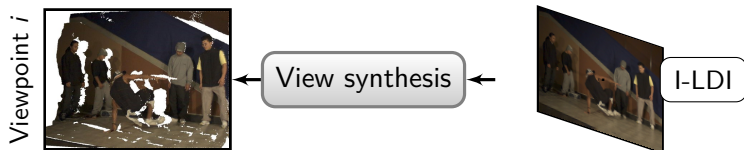


Compressed depth map

I-LDI: Incremental-LDI construction [Jantet et al., 3DTV]

Iterate for each input view

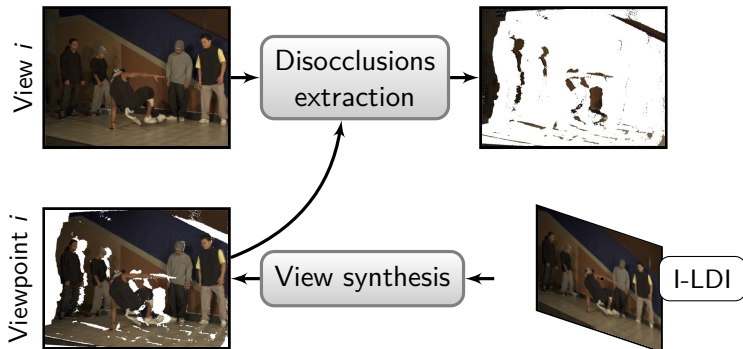
- Use current I-LDI to synthesize one acquired viewpoint
- Compare with captured view to compute disocclusion texture
- Insert back textures into the I-LDI



I-LDI: Incremental-LDI construction [Jantet et al., 3DTV]

Iterate for each input view

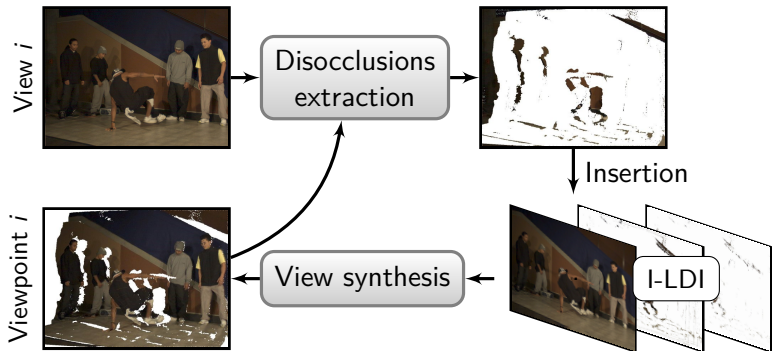
- Use current I-LDI to synthesize one acquired viewpoint
- Compare with captured view to compute disocclusion texture
- Insert back textures into the I-LDI



I-LDI: Incremental-LDI construction [Jantet et al., 3DTV]

Iterate for each input view

- Use current I-LDI to synthesize one acquired viewpoint
- Compare with captured view to compute disocclusion texture
- Insert back textures into the I-LDI



I-LDI vs LDI Comparison

LDI frames: many pixels in many layers, with scattered distribution



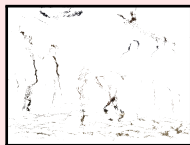
1st layer



2nd layer



3rd layer



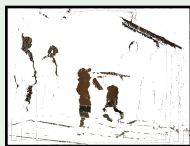
4th layer

...

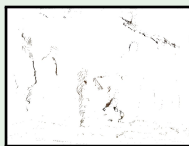
I-LDI frames: less pixels and less layers with compact distribution



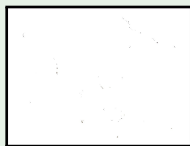
1st layer



2nd layer



3rd layer



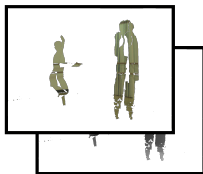
4th layer

...

Classical LDI limitations



1st layer



2nd layer

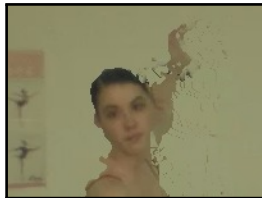


Compressed depth map

Compression artifacts

- Large depth discontinuities
- Motion in multi-layer
- Boundaries in multi-layer
- Uneasily compressed

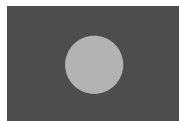
Introducing Object-based LDI



Synthesized virtual view

Depth compression artifacts

Organizes pixels into layers to enhance depth continuity



Visible layer



Occluded layer



Foreground



Background

Classical LDI depth layers

Object-based LDI depth layers

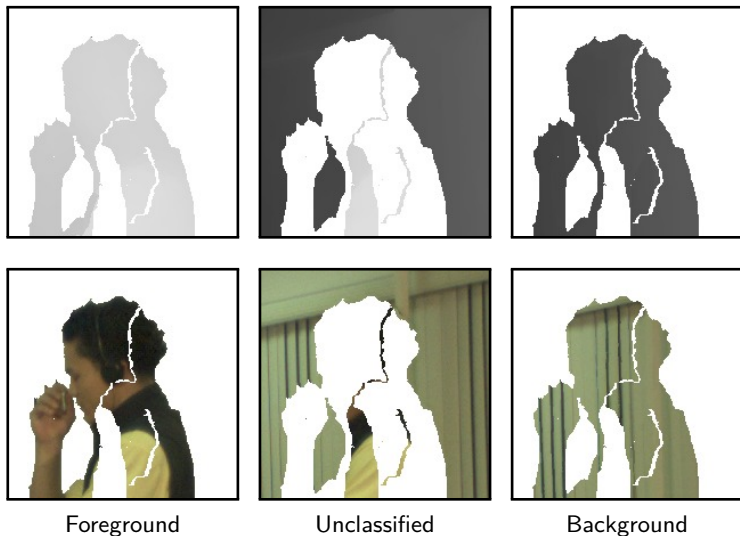
Method based on a region growing algorithm

Region R initialized with pixels where Z_{FG} and Z_{BG} are already defined

For each pixel q outside R :

- Extrapolate Z_{FG} and Z_{BG}
- Classify q

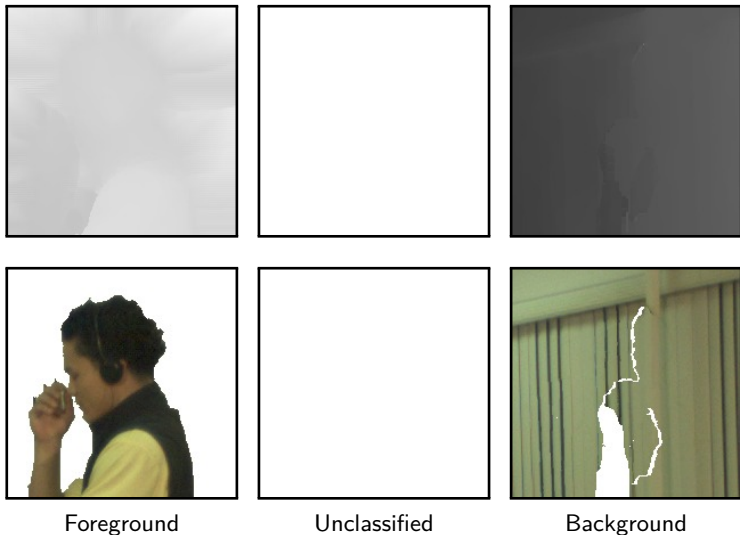
O-LDI: Classification Initializing



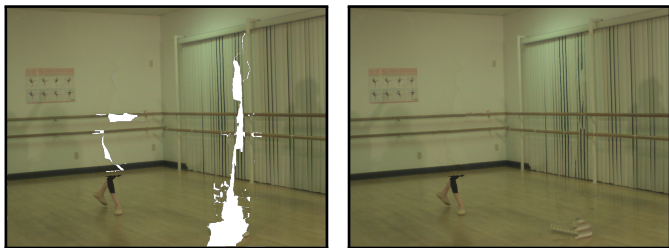
O-LDI: Classification Processing



O-LDI: Classification Results



O-LDI: Background inpainting



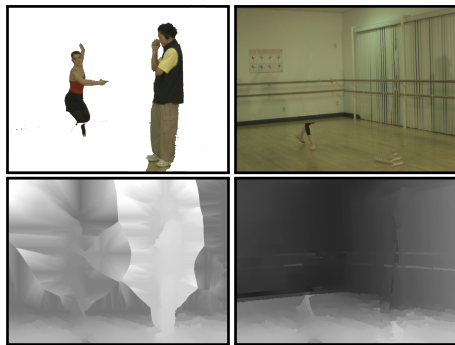
Background inpainting

Principe

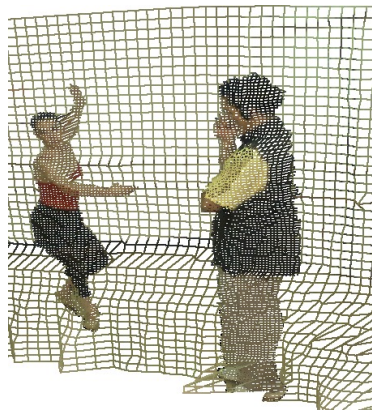
Exemplar-based inpainting from Criminisi [CPT03]

- Robust and time-consuming method
- Preserves texture and structure

O-LDI: Fast mesh-based rendering



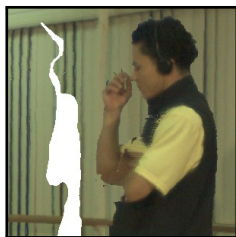
Object-based LDI



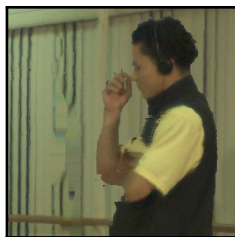
Meshes rendering

Continuous layers can be rendered as meshes
Foreground mesh is partially transparent

O-LDI: Rendering results



Disocclusions



Fast SoA inpainting



O-LDI rendering

Online inpainting limitations

Fast inpainting, introduces:

- Artifacts
- Stretching
- Temporal flickering

O-LDI advantages

Robust offline inpainting

- Time coherent rendering
- Multi-view coherent rendering

O-LDI Conclusions

O-LDI Advantages

- Static background along time
- Compatible with fast mesh-based rendering
- Depth continuity improves rendering quality
- Remove unnecessary boundaries \Rightarrow Should improve compression

O-LDI Limitations

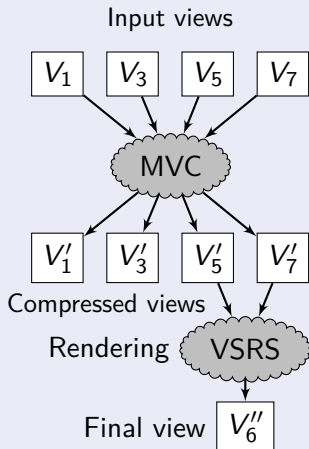
- No backward compatibility with 2D decoding scheme

Table of contents

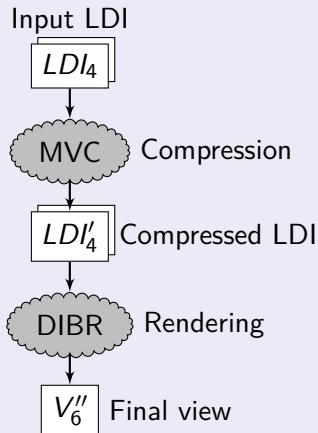
- 1 View synthesis (JPF)
- 2 Layered Depth Image (LDI)
- 3 LDI-based multi-view compression**
- 4 Conclusions

MVD and LDI compression schemes

MVD compression (MVC)



LDI compression (MVC)



"Breakdancing" multi-view video

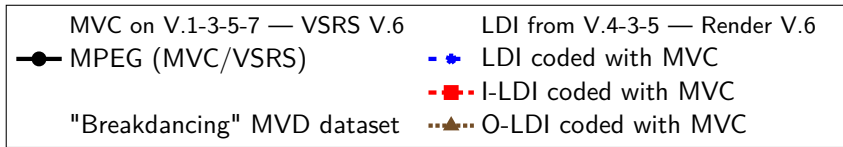
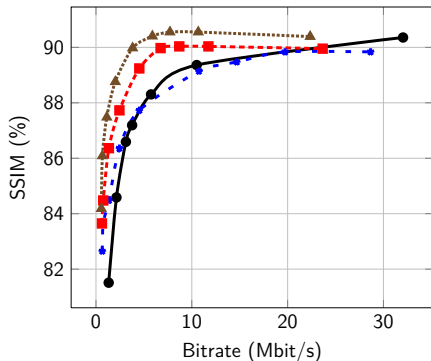
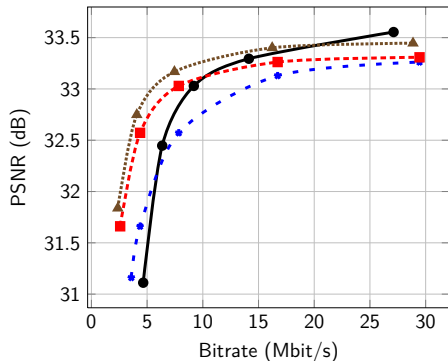


Table of contents

- 1 View synthesis (JPF)
- 2 Layered Depth Image (LDI)
- 3 LDI-based multi-view compression
- 4 Conclusions**

View synthesizing conclusions

JPF: Joint Projection Filling method

Projection with occlusion-compatible pixel scanning order

- Handles cracks
- Fills disocclusions with background
- Preserves geometrical structures

Virtual View Synthesis method with Full-Z Depth Aided Inpainting

First synthesizes virtual zMap to help synthesizing virtual view

- Preserves sharp boundaries
- Realistic disocclusions filling

Intermediate representation conclusions

I-LDI: Incremental Layered Depth Image

Iterative LDI construction to avoid layers correlations

- Less layers
- Less pixels
- Compact distribution

O-LDI: Object-based Layered Depth Image

Pixels reorganisation to enhance depth continuity

- Static background
- No depth discontinuities \Rightarrow No compression artifacts
- Compatible mesh-based rendering

Perspectives

Handle depth map inconsistencies

Non realistic depth maps drive down rendering quality

Improve temporal coherence

During LDI construction

During views projection

During Depth Aided Inpainting

Use more efficient compression scheme

Consider MPEG 3D-HEVC

Explore dedicated Depth Map Compression schemes

Publications

- [Bosc et al., 2010] Bosc, E., Jantet, V., Morin, L., Pressigout, M., & Guillemot, C. (2010). Vidéo 3d: quel débit pour la profondeur? In *CORESA*.
- [Bosc et al., 2011] Bosc, E., Jantet, V., Pressigout, M., Morin, L., & Guillemot, C. (2011). Bit-rate allocation for multi-view video plus depth data. In *3DTV*.
- [Jantet et al., 2011a] Jantet, V., Guillemot, C., & Morin, L. (2011a). Joint projection filling method for occlusion handling in depth-image-based rendering. *3D Research*, 2, 1–13.
- [Jantet et al., 2011b] Jantet, V., Guillemot, C., & Morin, L. (2011b). Object-based layered depth images for improved virtual view synthesis in rate-constrained context. In *ICIP*.
- [Jantet et al., 2009] Jantet, V., Morin, L., & Guillemot, C. (2009). Incremental-Idi for multi-view coding. In *3DTV*.
- [Jantet et al., 2010] Jantet, V., Morin, L., & Guillemot, C. (2010). Génération, compression et rendu de Idi. In *CORESA*.
- [Sourimant et al., 2009] Sourimant, G., Colleu, T., Jantet, V., & Morin, L. (2009). Recalage gps / sig / video, et synthèse de textures de bâtiments. In *CORESA*.
- [Sourimant et al., 2011] Sourimant, G., Colleu, T., Jantet, V., Morin, L., & Bouatouch, K. (2011). Toward automatic gis–video initial registration. *Annals of Telecommunications*, 67, 1–13.