Panoramic Video Texture

Aseem Agarwala, Colin Zheng, Chris Pal, Maneesh Agrawala,
Michael Cohen, Brian Curless, David Salesin, Richard Szeliski

A paper accepted for SIGGRAPH'05

presented by Markus Liechti

Outline

- Introduction & Motivation
- Problem Definition
- Panoramic Video Texture Algorithm
- Results
- Conclusion

Introduction: Panorama

- Normally done by stitching images together
- Reveals wide all-encompassing view
- Provides limited form of immersion

Robert Barker(1792)

PVT - Motivation

- Wide field of view
- Infinite, non repetitive length
- Immersive experience

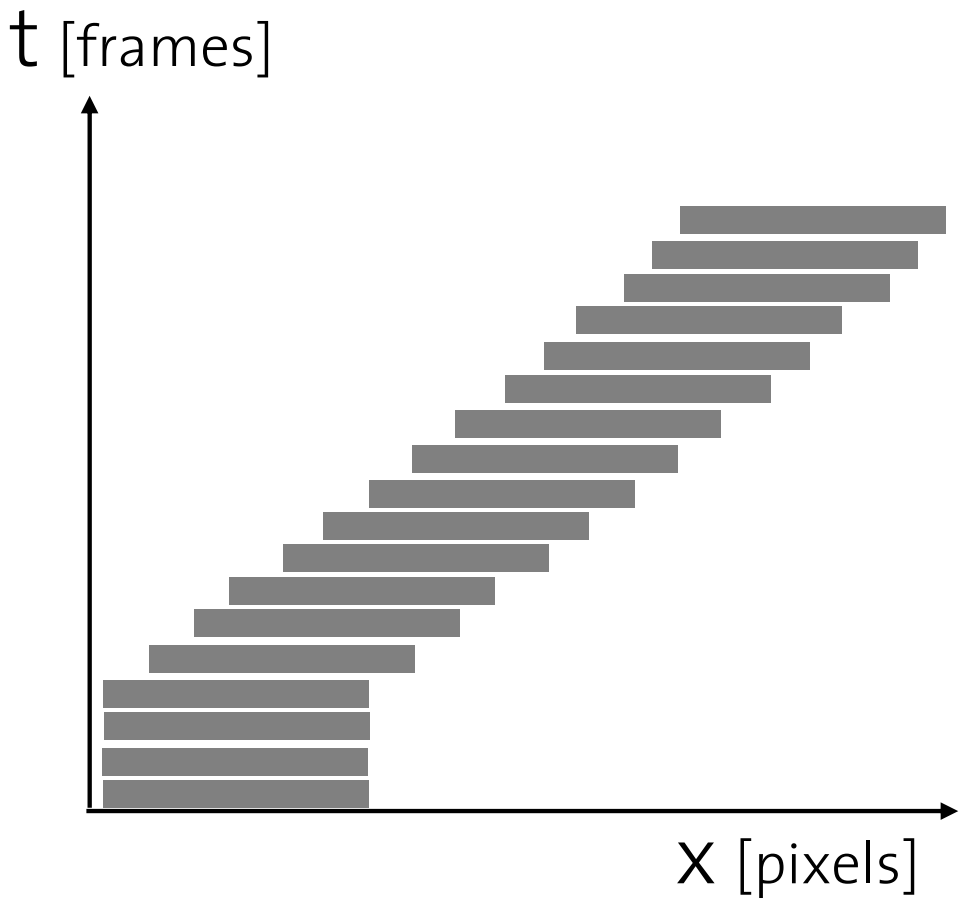
Related Works

- Image/video hybrids
  [Finkelstein 1996; Irani and Anandan 1998]
- Full video panoramas
  [Neumann 2000; Uyttendaele 2004]

- Drawbacks:
  – Finite duration in time
  – Need of specialized hardware
  – Restriction of resolution

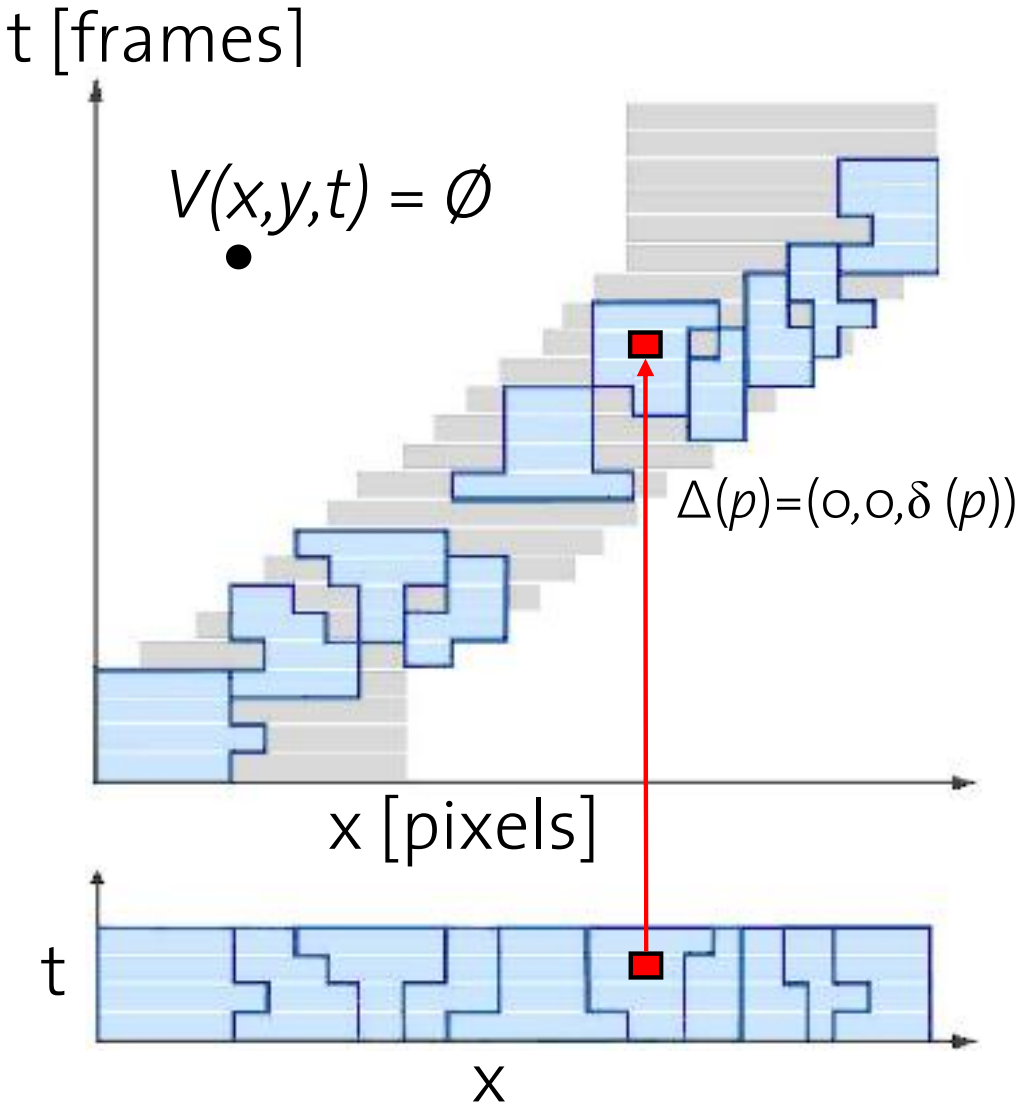
Input

Panning camera



Spatio-temporal volume

Notations



- Input volume: spatio-temporal volume $V(x,y,t)$
- *Output volume*
- Point/Pixel $p=(x,y,t)$
- Mapping

$$(x,y,t) \xrightarrow{\Delta(p)} (x, y, t + \delta(x,y,t))$$

Video Registration

- Align frames
- Warp video frames into one global coordinate system
- Problems with aperiodical movements
- Mask out moving parts of the input video

Dynamic & Static Regions

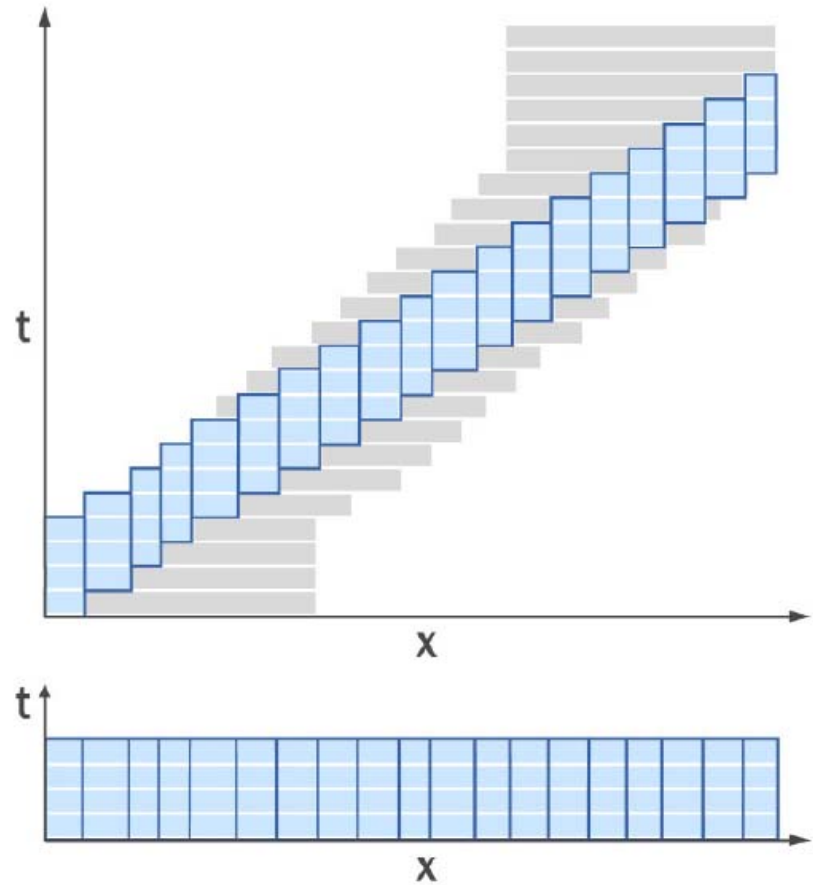
- Partion scene into separate dynamic and static regions
  - 1 frame for static part, saving memory
  - Dynamic regions can be computed independently and with different durations

Dynamic & Static Regions

- Partition scene into separate dynamic and static regions
  - 1 frame for static part, saving memory
  - Dynamic regions can be computed independently and with different durations
- Background layer *B(x,y)*
- Binary matte *D* for dynamic regions
  - *D* overlaps non-null regions of *B* along one-pixel wide boundary

Simple Approach

- Sheared rectangular slice through input volume
- Shear it into the output volume.
- May change structure of motion in scene



Simple Approach

Problem Definition

Given a finite segment of video V shot by a single panning camera, create a mapping $\Delta(p) = (0,0,\delta(p))$ for every pixel p in the output panoramic video texture, such that $V(p+\Delta(p)) \neq \emptyset$ and the *seam cost* of the mapping $C_s(\Delta)$ is minimized.

Seam cost (1)

$$C_s(\Delta) = \sum_{p=(x,y,t)\,|\,(x,y)\in D} \left(C_b(\Delta, p) + C_v(\Delta, p) \right)$$

Boundary cost

$$C_b(\Delta, p) = \begin{cases} \|V(p+\Delta(p)) - B(p)\|^k & \text{if } B(p) \neq \emptyset; \\ 0 & \text{otherwise.} \end{cases}$$

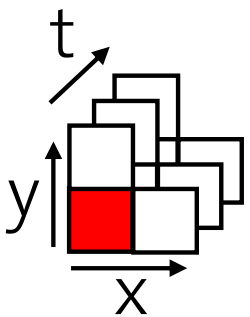
Volume cost

$$C_v(\Delta, p) = \sum_{i=1}^{3} \begin{cases} \|V(p+\Delta(p)) - V(p+\Delta(p+e_i))\|^k & \text{if } p+e_i \in D; \\ 0 & \text{otherwise.} \end{cases}$$

Seam cost (2)

$$C_v(\Delta, p) = \sum_{i=1}^{3} \begin{cases} \|V(p+\Delta(p)) - V(p+\Delta(p+e_i))\|^k & \text{if } p+e_i \in D; \\ 0 & \text{otherwise.} \end{cases}$$

- Cost function C_s maps onto 3D Markov random field (MRF)
  - Domain $D \times [0..t_{max}]$
  - Free variables $\Delta(p)$ resp. $\delta(p)$



Typical values

- Size of aligned input video
  - 6000 x 1200 pixels spatially
  - 1000's of frames
  - 500 choices for δ (p) for every pixel p
- Output video
  - 35 frames
- Output volume
  - 6000 x 1200 x 35 ≈ 2.5 x 10^8 variables each with 500 possible values

PVT algorithm

1. Preprocessing steps
   - Video registration using existing techniques
   - User-drawn mask to separate static and dynamic regions
   - Determine looping length
2. Constrained formulation
3. Loosening constraints
4. Hierachical min-cut optimization
5. Gradient-domain compositing

Algorithm: Looping length

- Choose l_{min} and l_{max}
- Determine best loop length t_{max}
  - Compare each input frame t to each input frame in the range (t+ l_{min}, t+ l_{max}) that spatially overlaps with frame t by at least 50%.
- Find pair of frames t, $t+l$ and set $t_{max}(t)$ to l-1

PVT algorithm

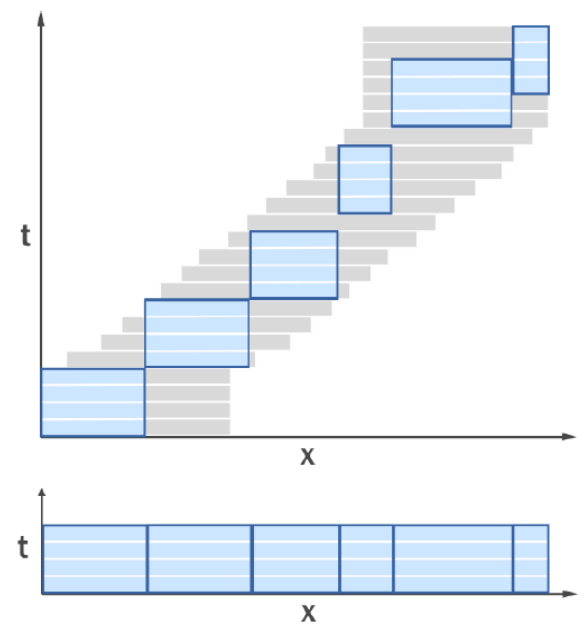
1. Preprocessing steps
   - Video registration using existing techniques
   - User-drawn mask to separate static and dynamic regions
   - Determine looping length
2. Constrained formulation
3. Loosening constraints
4. Hierachical min-cut optimization
5. Gradient-domain compositing

Alogrithm: Constrained Formulation

1. Single time offset δ at each output location *(x,y)*, regardless of *t*
2. Set $\Delta(p)$ the same for each pixel in a given column of the output

→ 1D search space

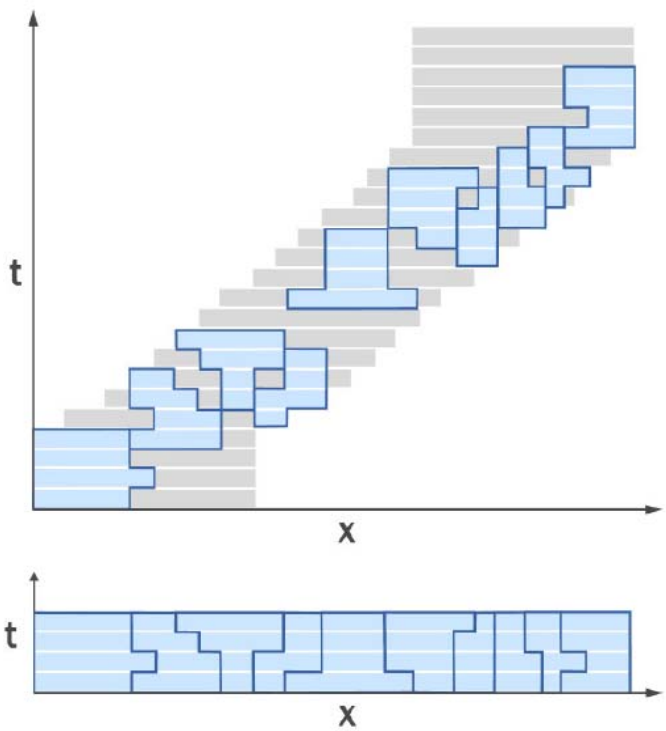
→ List of time offsets



m different time offsets

Alogrithm: Loosening the constraints

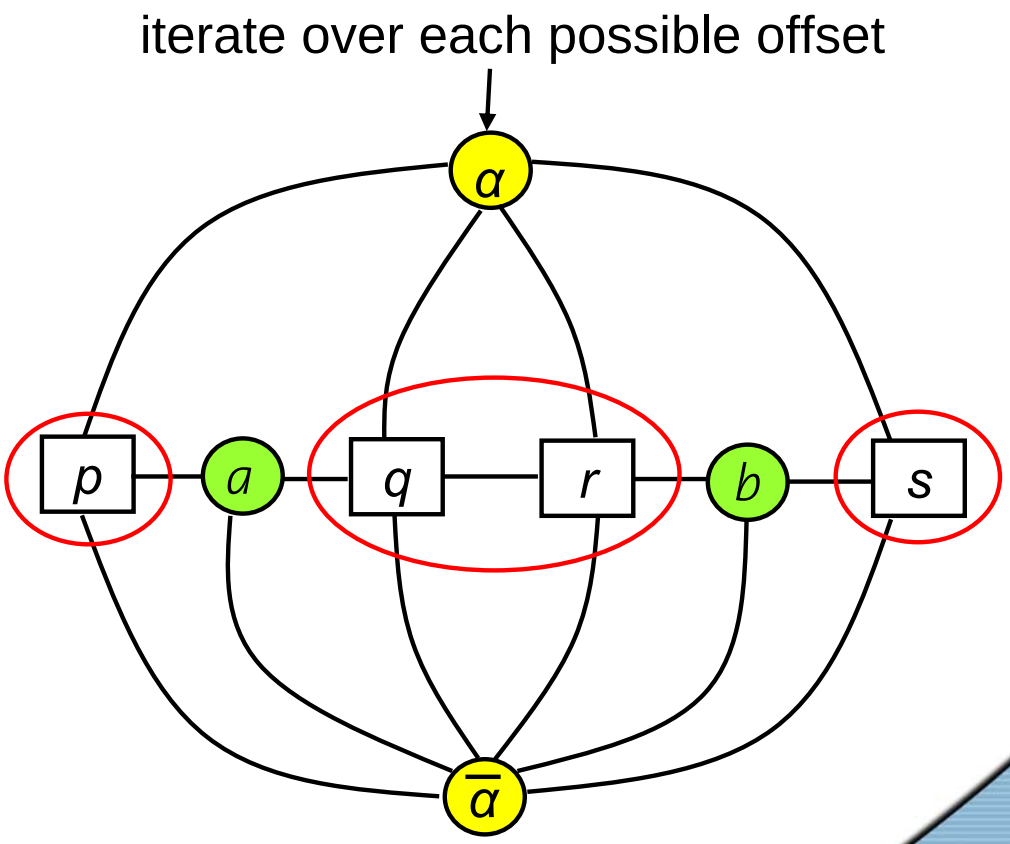
- Consider now the full 3D MRF Problem
- Every pixel may take on different mapping $\Delta(p) = (0,0,\delta(p))$
- *$\delta(p)$ restricted to 2m time offsets*



2m different time offsets

Loosening Contraints
Iterative min-cut optimization

- Define 3D-graph with each node representing one pixel in output video volume
- Group pixels with same offset
- Still too high computational costs



1D Representation

Alogrithm: Hierarchical min-cut optimization

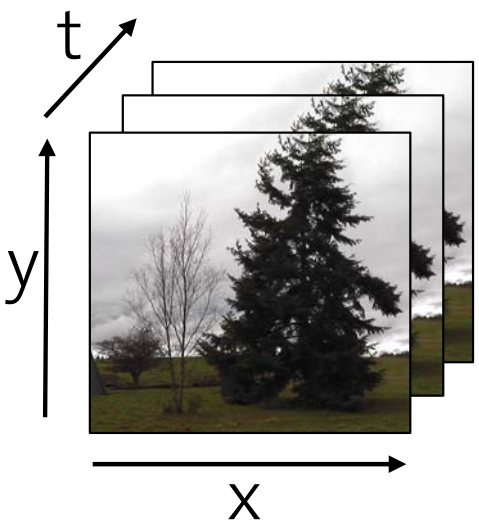
- Compute solution at coarser resolution
- Seams at finer resolution will be roughly similar to those at coarser resolution
- Approach finer resolution using 2-3 hierachical steps

PVT algorithm

1. Preprocessing steps
   - Video registration using existing techniques
   - User-drawn mask to separate static and dynamic regions
   - Determine looping length
2. Constrained formulation
3. Loosening constraints
4. Hierachical min-cut optimization
5. Gradient-domain compositing

Algorithm: Gradient-domain compositing

- Visual artifacts
  - Exposure variations
  - Errors in alignment procedure
- Treat frames as 3D volume cube
- Composit video in the gradient domain
- Treat video as source of color gradients
- Integrate 3D gradient field to create video texture



Results

7400 x 1300, 107MB

← 2600 x 1400, 106MB

3300 x 800, 65MB

Performance

- 2-7 hours to compute
- Performance loss due to swapping video frames in and out of memory
- More intelligent frame caching would reduce time

Limitations

- Scene needs some kind of stationarity
- Aperiodical movements
- Overlapping of periodically moving elements
- Failure to observe a complete cycle of a periodic phenomenon

Future Work

- Automatic segmentation of the scene into dynamic and still regions
- Add audio
- Video sprites
- 3D

Discussion