Multi Viewpoint Panoramas

Stefan Elstner

27. November 2007
1 Motivation

2 Methods
   - Slit-Scan
   - "The System"

3 "The System"
   - Approach
   - Preprocessing
   - Surface Selection
   - Panorama Creation
   - Interactive Refinement

4 Sources

Stefan Elstner
Multi Viewpoint Panoramas
Motivation

- image showing long continuous scene
- single image not sufficient
  - only small part of street
  - greater field of view: distortions
  - from far away: not always possible, loss of details
- solution: capture multiple images from different points of view
- -> method needed to stitch images together
Slit-Scan Overview

- "strip panoramas"
- ancient: slit shaped aperture across film
- today: thin vertical strips of pixels of video source
- orthographic projection along horizontal axis
- perspective projection along vertical axis
Disadvantages

- distortions of objects not in specific distance from camera plane
  - farther away: horizontally stretched
  - closer: squashed
- taken from video footage
  - poor quality
Examples

- Example 1 - Downtown L.A. MVP by Dietmar Offenhuber

- Example 2 - Artistic usage
"The System" - Overview

- inspired by work of Michael Koller
- goal: reduce disadvantages of strip panoramas
- composites arbitrary regions of the source images
- Markov Random Field (MRF) optimization
- allows interactive refinement
Approach 1

- properties of "well-visualizing" multi viewpoint panoramas
  - each object in scene rendered from viewpoint roughly in front of it
  - composed of regions seen from natural point of view, linear perspective
  - objects closer to image plane larger than further away objects
  - seams between perspective regions do not draw attention
Approach II

- steps to multi viewpoint panorama
- source images:
  - handheld photographs
  - auto focus
  - manual exposure
Approach III

- plan view of hypothetical scene
- geometry lying along large dominant plane
- images projected onto picture surface from original 3D viewpoints agree in areas describing scene geometry on dominant plane

**Visualization example**
Preprocessing

- removal of radial distortions (e.g. when fish eye lens used)
- recovery of projection matrices of each camera \( i \)
  - 3D rotation matrix \( R_i \)
  - 3D translation matrix \( t_i \)
  - focal length \( f_i \)
  - camera location in world coordinates: \( C_i = -R_i^T t_i \)
- structure-from-motion system
  - matches SIFT features
- compensate exposure variations between source images
  - brightness scale factor \( k_i \), least squares in matching SIFT points between pairs of images \( l_i \) and \( l_j \)
  - \( k_i l_i = k_j l_j \)
Surface Selection I

- picture surface selected by user
  - should be roughly aligned with dominant plane
  - will be extruded in y dimension
- aid: view of recovered 3D points

- blue line: picture surface selected by user
- red dots: extracted camera locations
Surface Selection II

- automatic definition of coordinate system
  - fitting plane to camera viewpoints using PCA
- interactive definition of coordinate system by user
  - two points form x axis
  - two points form y axis
  - cross product results in z axis
  - cross product of z and y then form new x axis
Surface Selection III

- easy to identify dominant plane
- little harder to identify dominant plane
Surface Selection IV

- projection of source images onto picture surface

- $S(i,j)$ describes 3D location of sample $(i,j)$ on picture surface

- samples $S(i,j)$ are projected into source photographs

- result for one image
Viewpoint Selection

- each image $l_i$ represents $i^{th}$ viewpoint
- equivalent dimensions
- choose color for each pixel $p = (p_x, p_y)$ in panorama from one source image: $l_i(p)$
- objective function
  - minimized using MRF optimization
  - labeling of each pixel: $L(p) = i$
Objective Function Term I

- object in scene rendered from viewpoint roughly in front of it
- vector starting at $S(p)$ of picture surface
- extending in direction of normal of picture surface
- angle between $C_i - S(p)$ and above vector
- the higher the angle the less in front of object
- simpler approach (approximation)
  - find pixel $p_i$ closest to camera $C_i$
  - 2D distance from $p_i$ to $p_{L(p)}$
- $D(p, L(p)) = |p - p_{L(p)}|$
Objective Function Term II

- natural and seamless transitions between different regions of linear perspective
  - look at pairs of neighbouring pixels $p$ and $q$
  - $V(p, L(p), q, L(q)) = |I_{L(p)}(p) - I_{L(q)}(p)|^2 + |I_{L(p)}(q) - I_{L(q)}(q)|^2$
Objective Function Term III

- resemble average image where scene geometry intersects picture surface
  - to some extent occurring naturally
  - problems: motion, specular highlights, occlusions
- mean and standard deviation for each pixel $p$
- vector median filter across color channels $\rightarrow$ robust mean
- median absolute deviation $\rightarrow$ robust standard deviation
- if $\sigma(p) < 10$ (color channels from 0 to 255)
  - $H(p, L(p)) = |M(p) - I_{L(p)}(p)|$
- otherwise
  - $H(p, L(p)) = 0$
Objective Function

\[ \sum_p (\alpha D(p, L(p)) + \beta H(p, L(p))) + \sum_{p,q} V(p, L(p), q, L(q)) \]

- \( L(p) \) not allowed if camera i does not project to pixel p
- form of Markov Random Field
- min-cut optimization

- \( \alpha \) and \( \beta \) determined experimentally (\( \alpha = 100, \beta = 0.25 \))
- higher \( \alpha \): more straight on views but more noticeable seams
- lower \( \alpha \) and \( \beta \): removal of objects off the dominant plane (power line, cars)
Summary

- Source photographs
- Projected sources
- Average image
- Final result ... Seams visualized
- Final result

- computing time reduced by computing at lower resolution first
- higher resolution versions created using hierarchical approach
- final panorama calculated in gradient domain (smooth errors across seams)
Interactive Refinement Overview

- user might not like result (seams etc.)
- interactive control over the resulting panorama
Seam Suppression

no seam should not cross stroke

result taking user interaction into account

stroke propagated from one source image to all others by using 3D knowledge
Interactive Refinement

**View Selection**

- user selects one source image
- user selects location where source image should appear in final panorama by doing strokes

selected location

result taking user interaction into account

Stefan Elstner
Multi Viewpoint Panoramas
Sources

