視覚情報処理論　（学環）
Visual Information Processing

コンピュータビジョン　（情・電子情報）
Computer Vision

三次元画像処理特論　（情・ｺﾝﾋﾟｭｰﾀ科学）
Three-Dimensional Image Processing

2013/12/11（水）16:40-18:10

池内 克史（大学院情報学環 教授）

代理：
小野 晋太郎
（生産技術研究所 特任准教授、博士（情報理工学））

東京大学
THE UNIVERSITY OF TOKYO

Time-varying Image Processing

- **Introduction**
- **Basic technologies**
  - Background subtraction
  - Optical flow
  - Structure from Motion (SfM)
  - Space-time Image Analysis
- **Applied technologies**
  - Introducing recent research cases

This time

Next time

Applied Technologies

東京大学
THE UNIVERSITY OF TOKYO

Video
Restoration & Summarization

- **Space-time Completion of Video (Y. Wexler)**
  - Completing deficits in video (based on color patch)
- **Motion Field Transfer (T. Shiratori)**
  - Completing deficits in video (based on optical flow)
- **Full-Frame Video Stabilization (Y. Matsushita)**
  - Restoring motion blurs in video
- **Space-time Video Montage (H.W. Kang)**
  - Summarization

Jointing Videos
(in spatial, & in temporal)

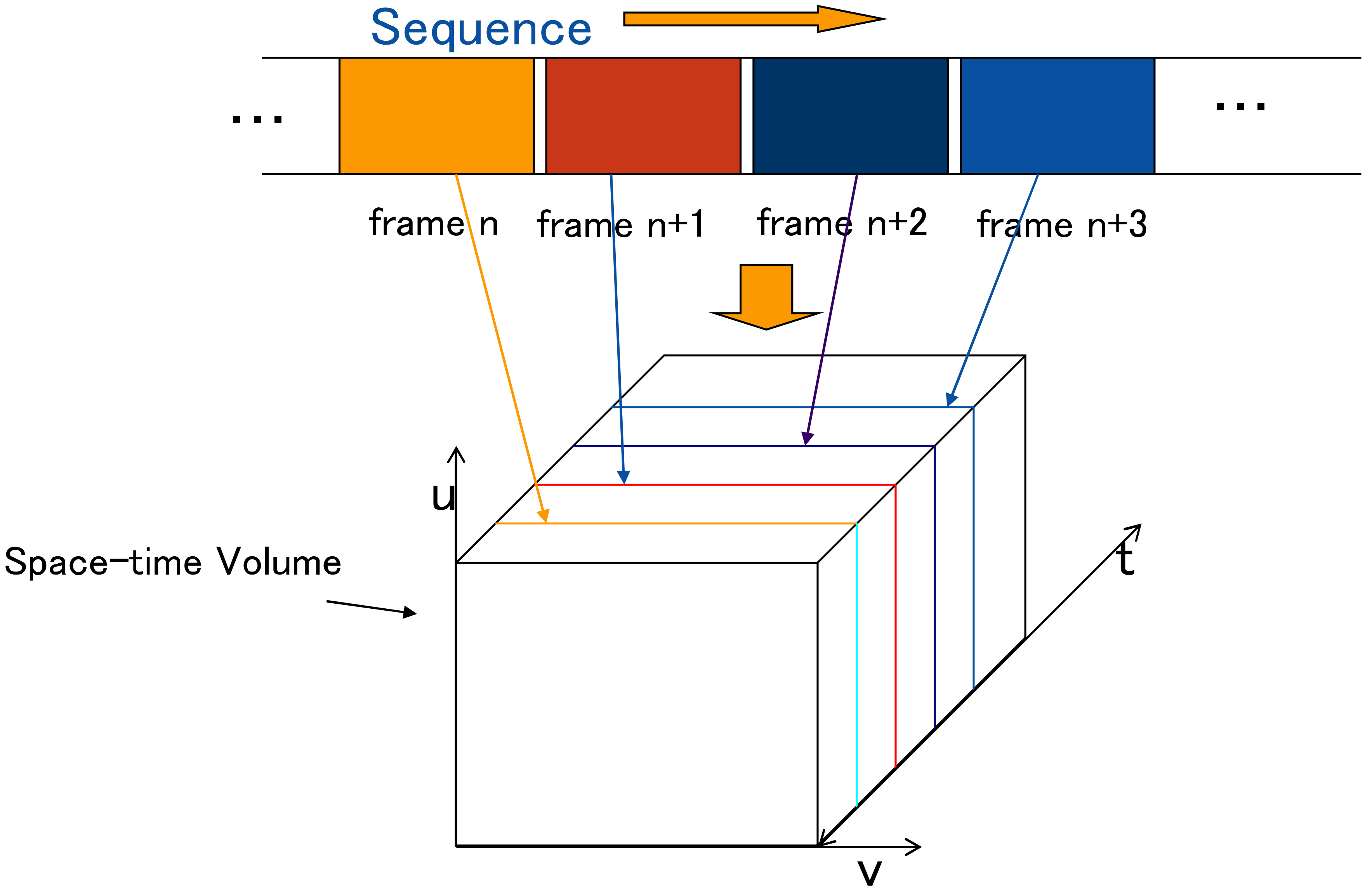
- **Video Textures (A. Schodel)**
  - Temporal joint
  - Creating infinite loop video by jointing similar frames
- **Aligning Non-Overlapping Sequences (Y. Caspi)**
  - Spatial joint
  - Relative position/pose between the videos are fixed
- **Video Matching (P. Sand)**
  - Spatial joint
  - Matching two videos looking same sequences, but captured in different opportunities

Others

- **Space-Time Behavior Based Correlation (E. Shechtman)**
  - Finding similar behaviors in videos based on gradient
- **Detecting Irregularities in Images and in Video (O. Boiman)**
- **Motion Magnification (C. Liu)**
  - Magnifying motions in a video
- **Space-Time Super-Resolution (E. Shechtman)**
  - Raising resolution of a video, regarding the frames as affine transformation in a space-time volume
- **Absolute-Scale SfM (Scaramuzza)**

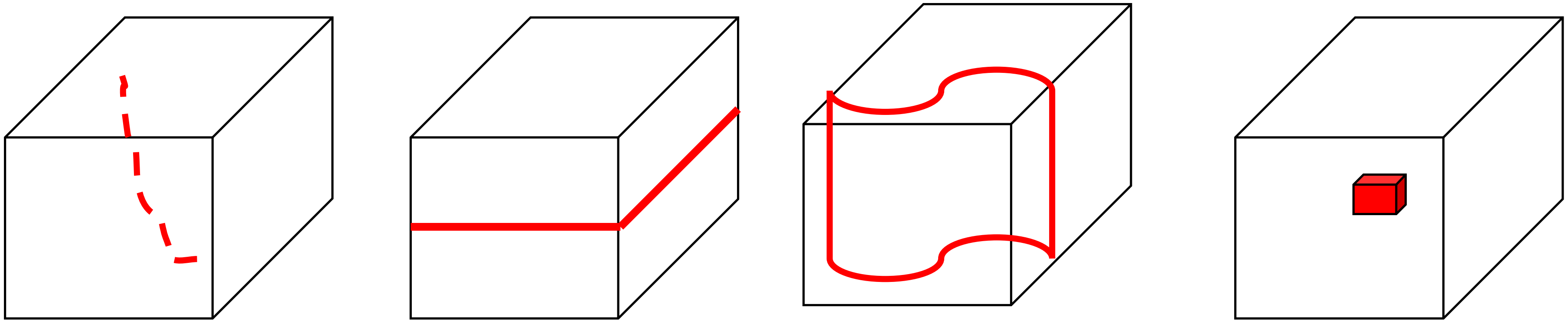
Space-time Volume

Space-time Volume



Information from Space-Time Volume

Use partial information



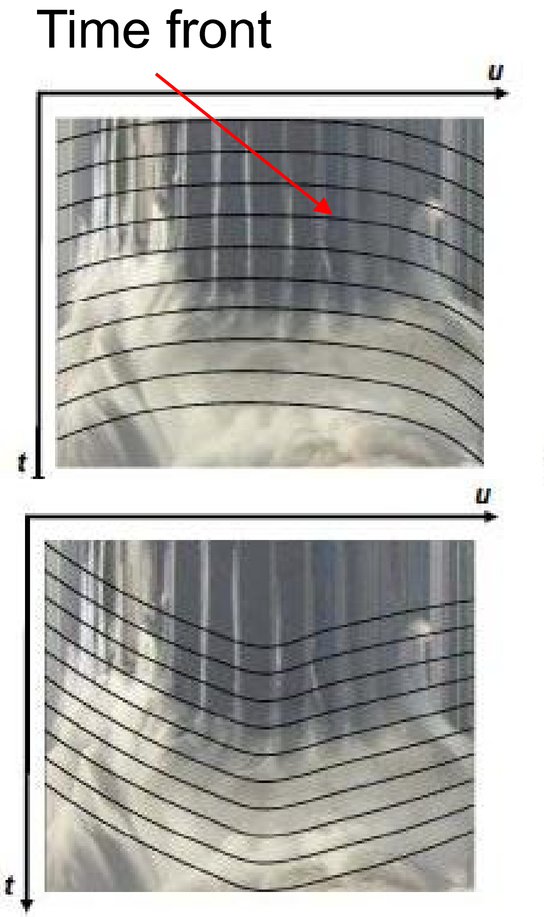
Trajectory

Slice

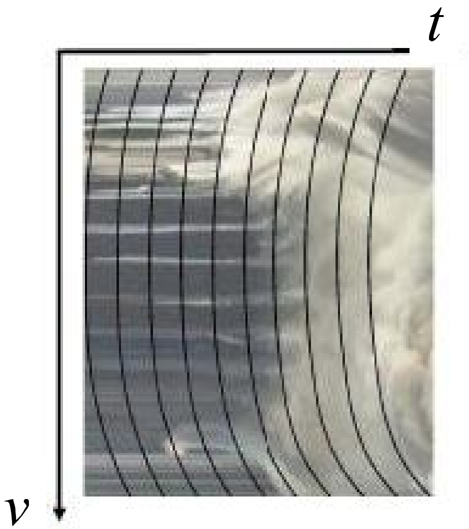
3D-block/shape

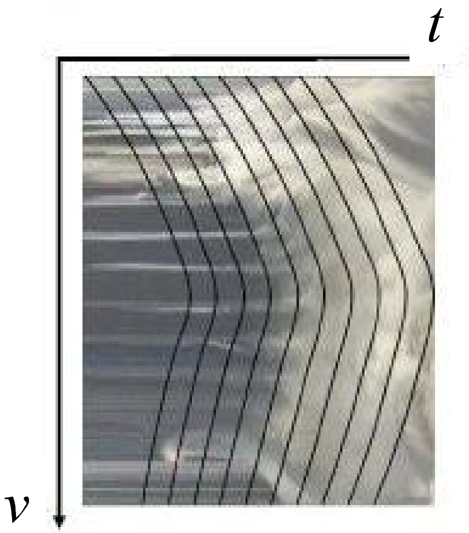
Temporal Video Editing
(Peleg 2005)

- The same original video with different time flow
- Show result image along time front slice



Temporal Video Editing

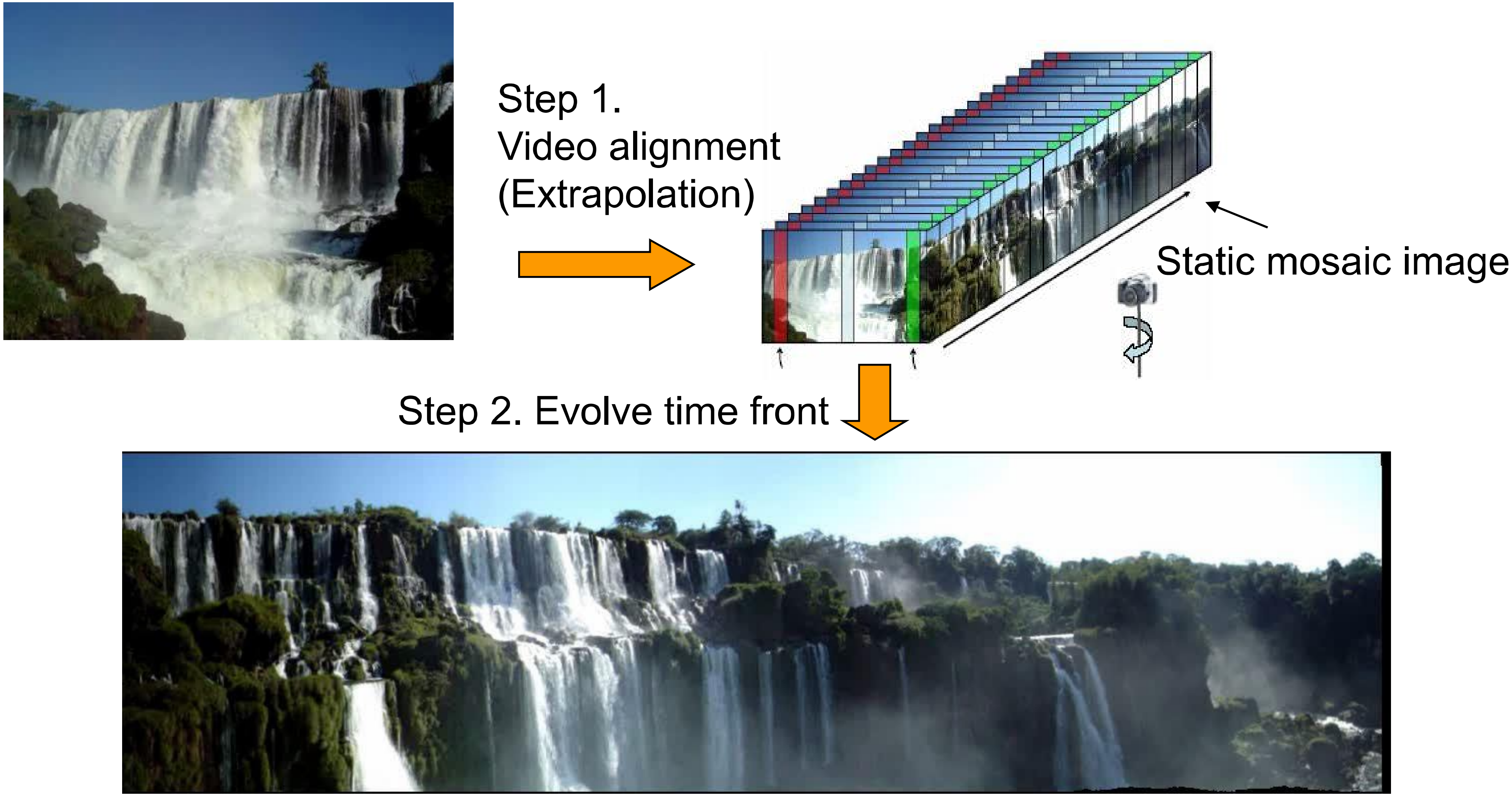




Rigging a Swimming
Competition

Dynamic Mosaics
(Peleg 2007)

One moving video camera is capturing a dynamic scene

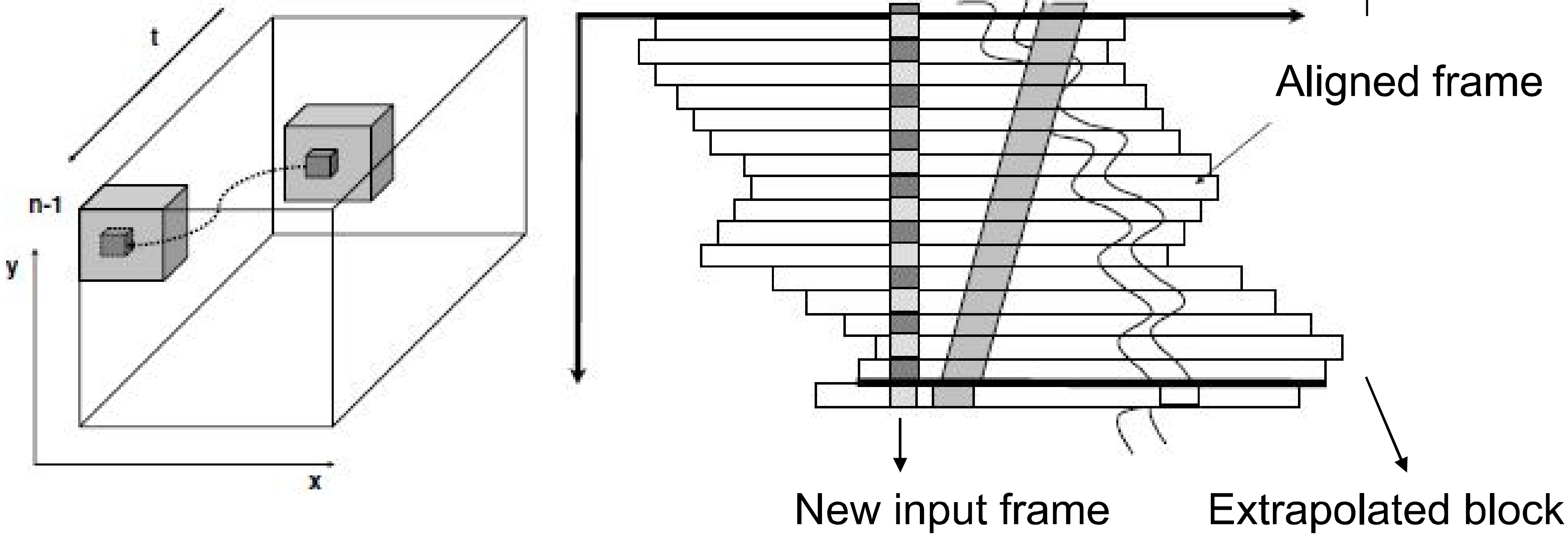


Step 1.
Video alignment
(Extrapolation)

Static mosaic image

Step 2. Evolve time front

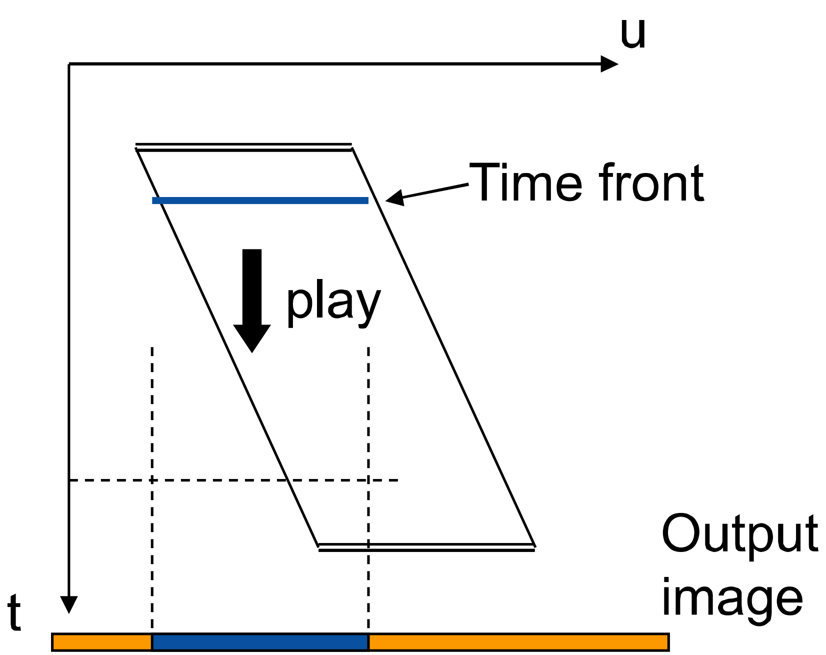
Dynamic Mosaics(2)
Step 1. Video Alignment (Extrapolation)



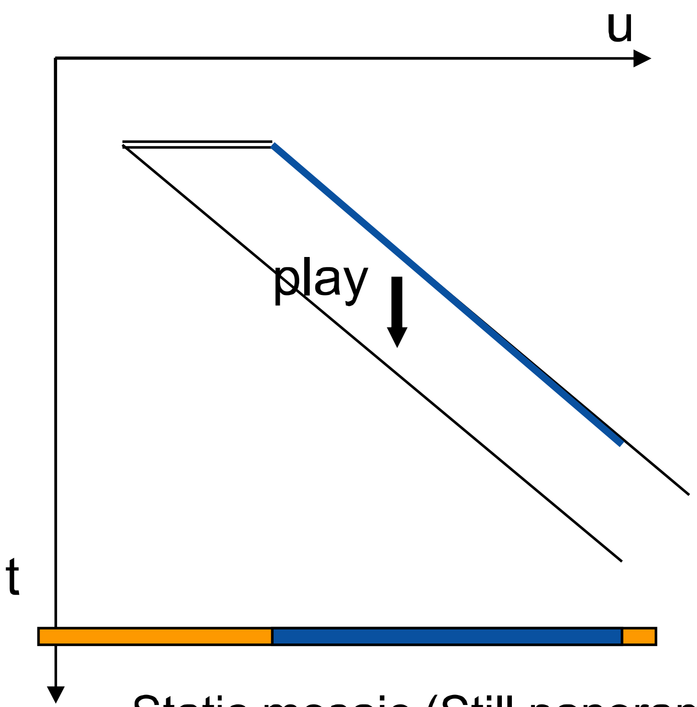
- Search similar blocks by SSD (sum of square differences)
- New frame can be extrapolated by past corresponding 3D-blocks
- Estimate the homography between new extrapolated frame and new input frame
- New input frame is aligned!!

Dynamic Mosaics(3)
Step 2. Evolve time front

Normal Video

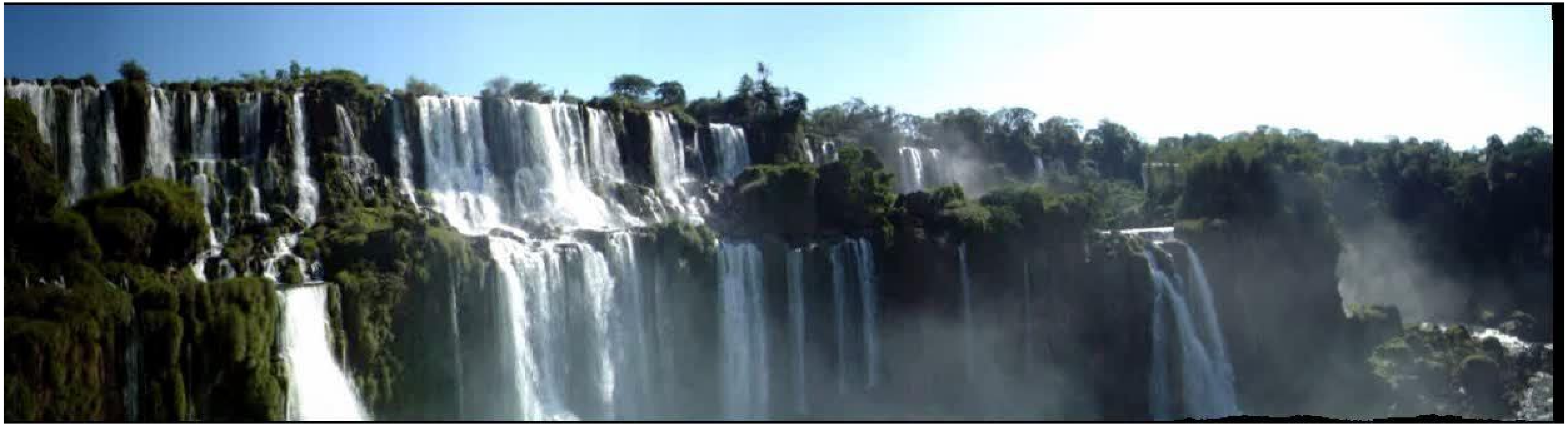


Mosaic Video

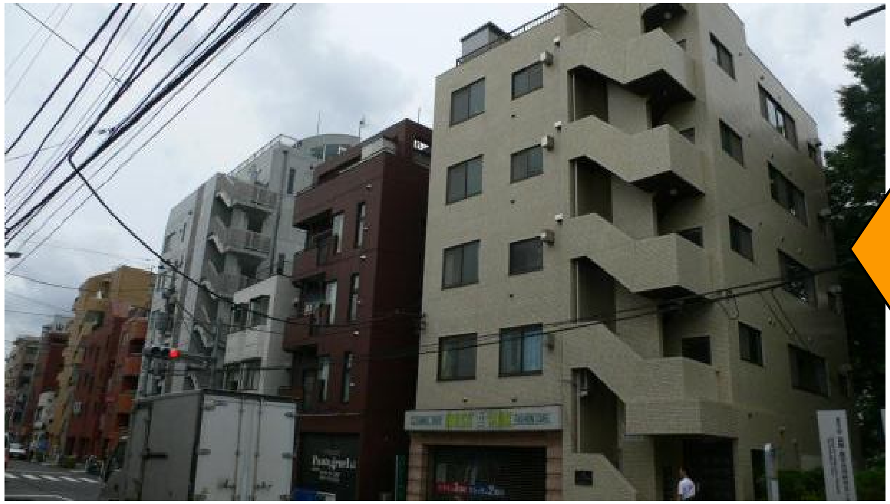


Static mosaic (Still panorama)

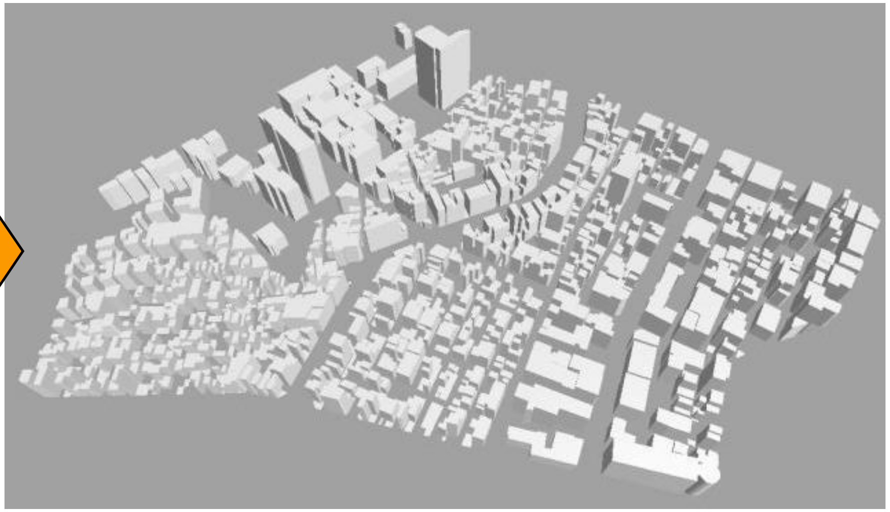
Result



Spacetime Feature Matching for Texturing (Wang 2008)



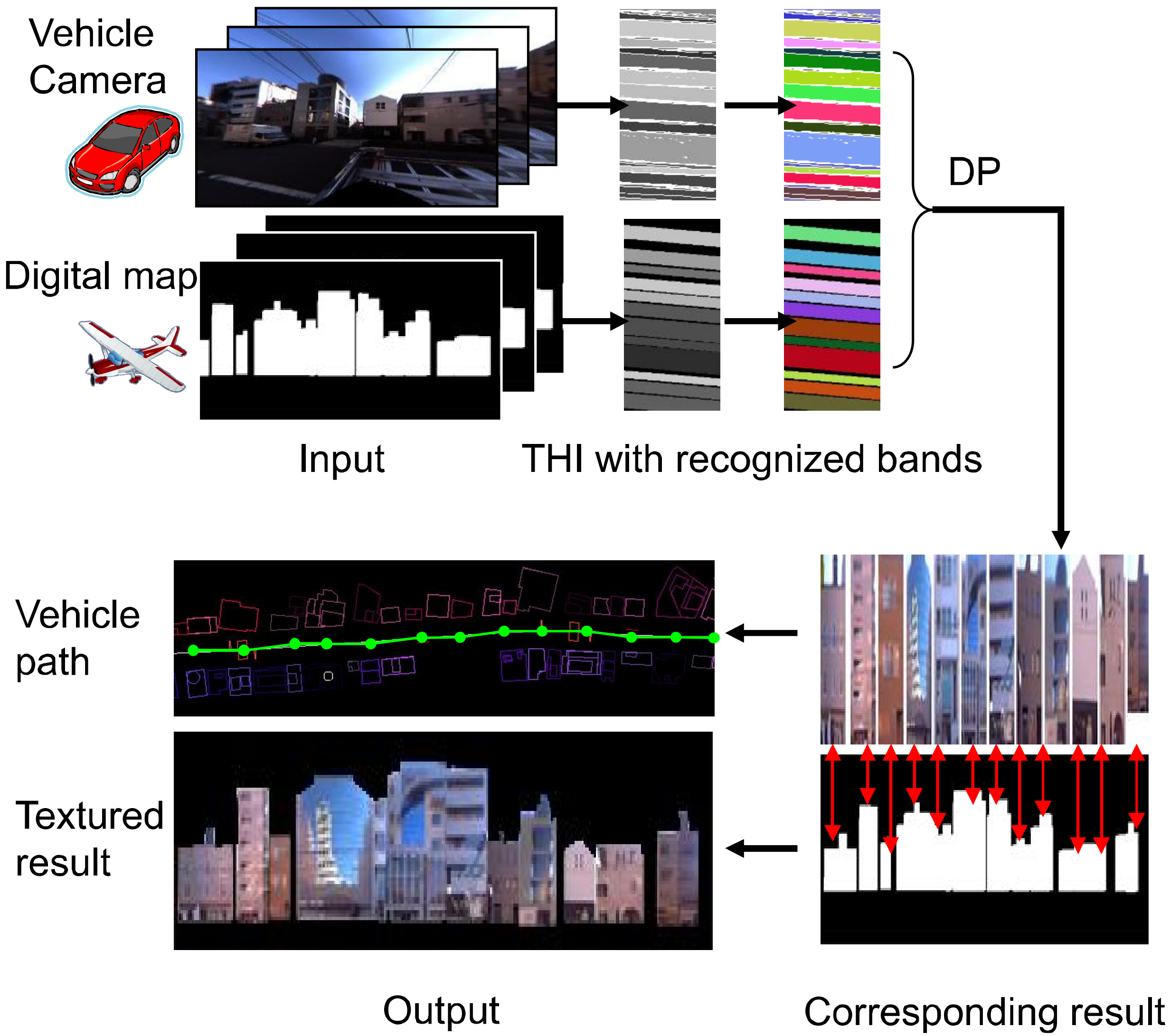
Ground-view image
(Vehicle survey, Local)



3D residential map
(Aerial survey, Global)

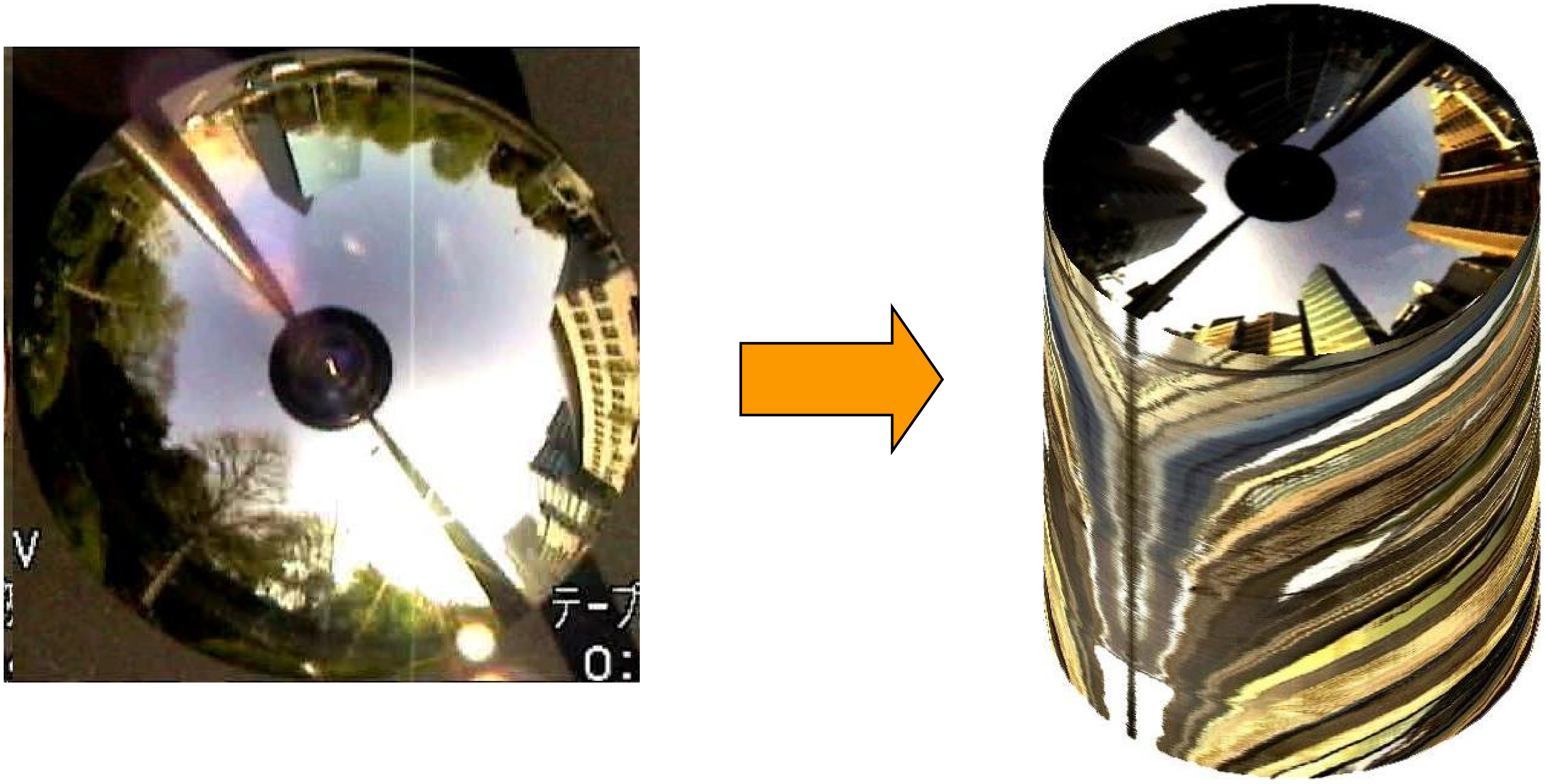
How can we get correspondence, and add a texture onto building walls?

Spacetime Feature Matching for Texturing (Wang 2008)

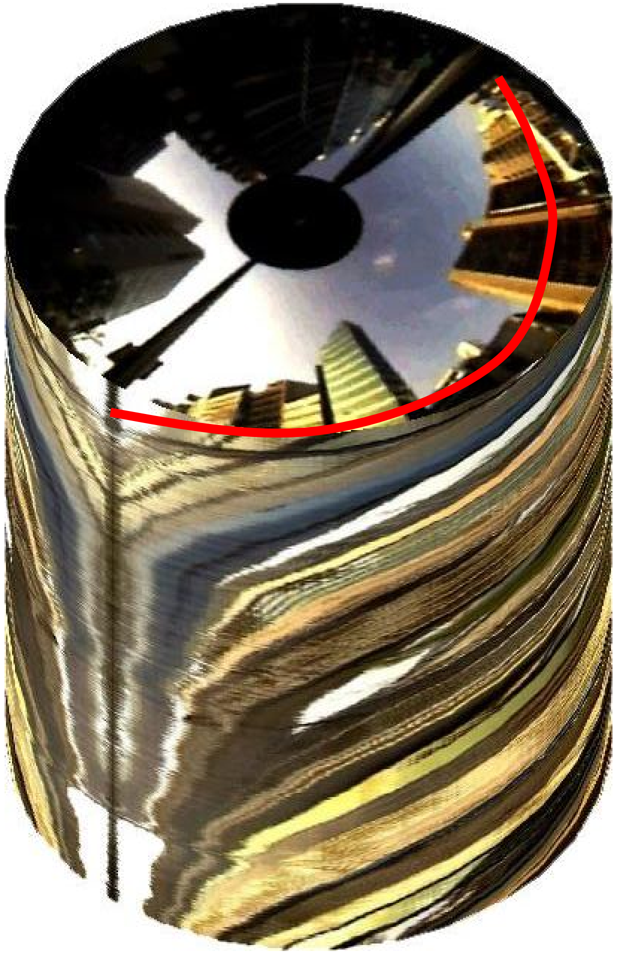


Omnidirectional Camera

Spatio-temporal volume of omni-directional image

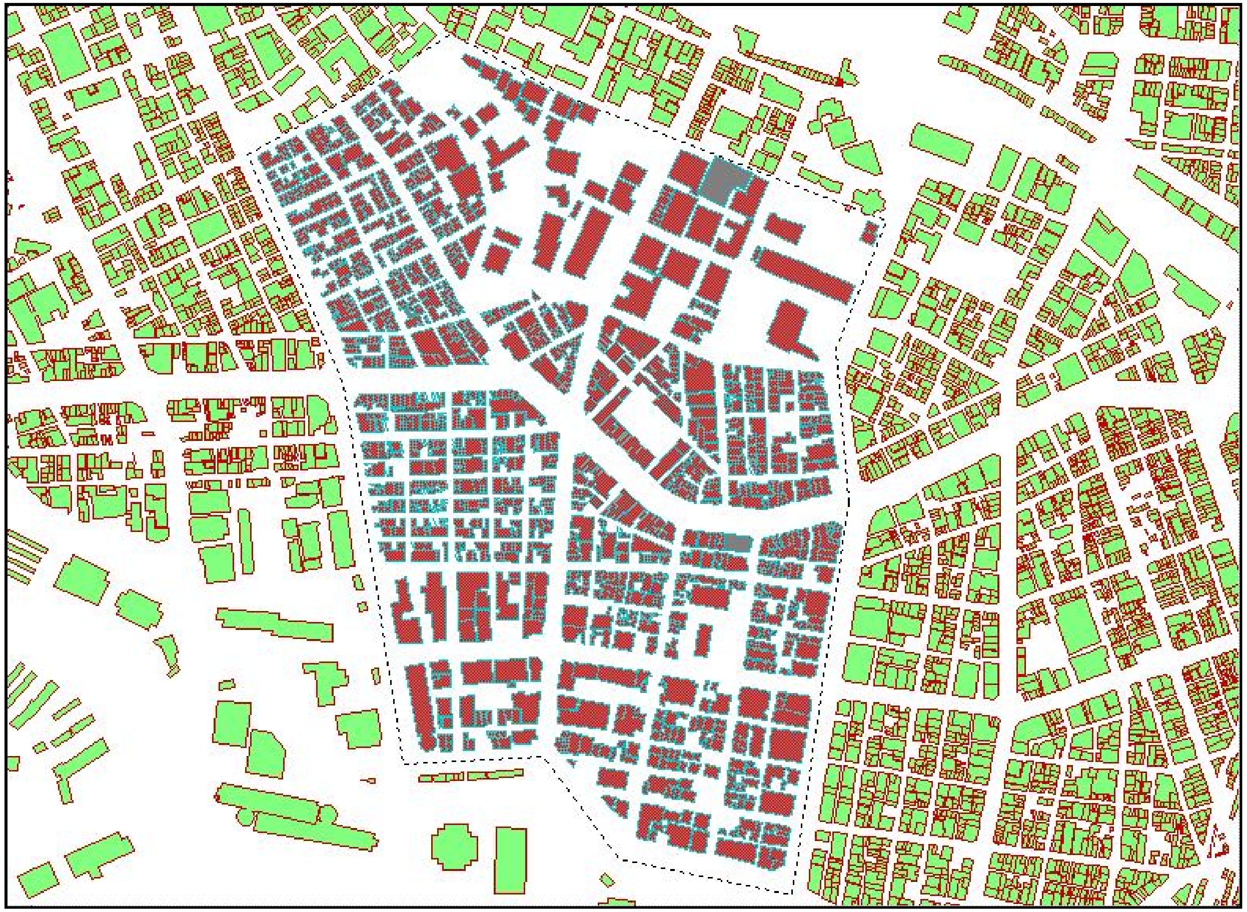


Cross-section
(an elliptic curve)



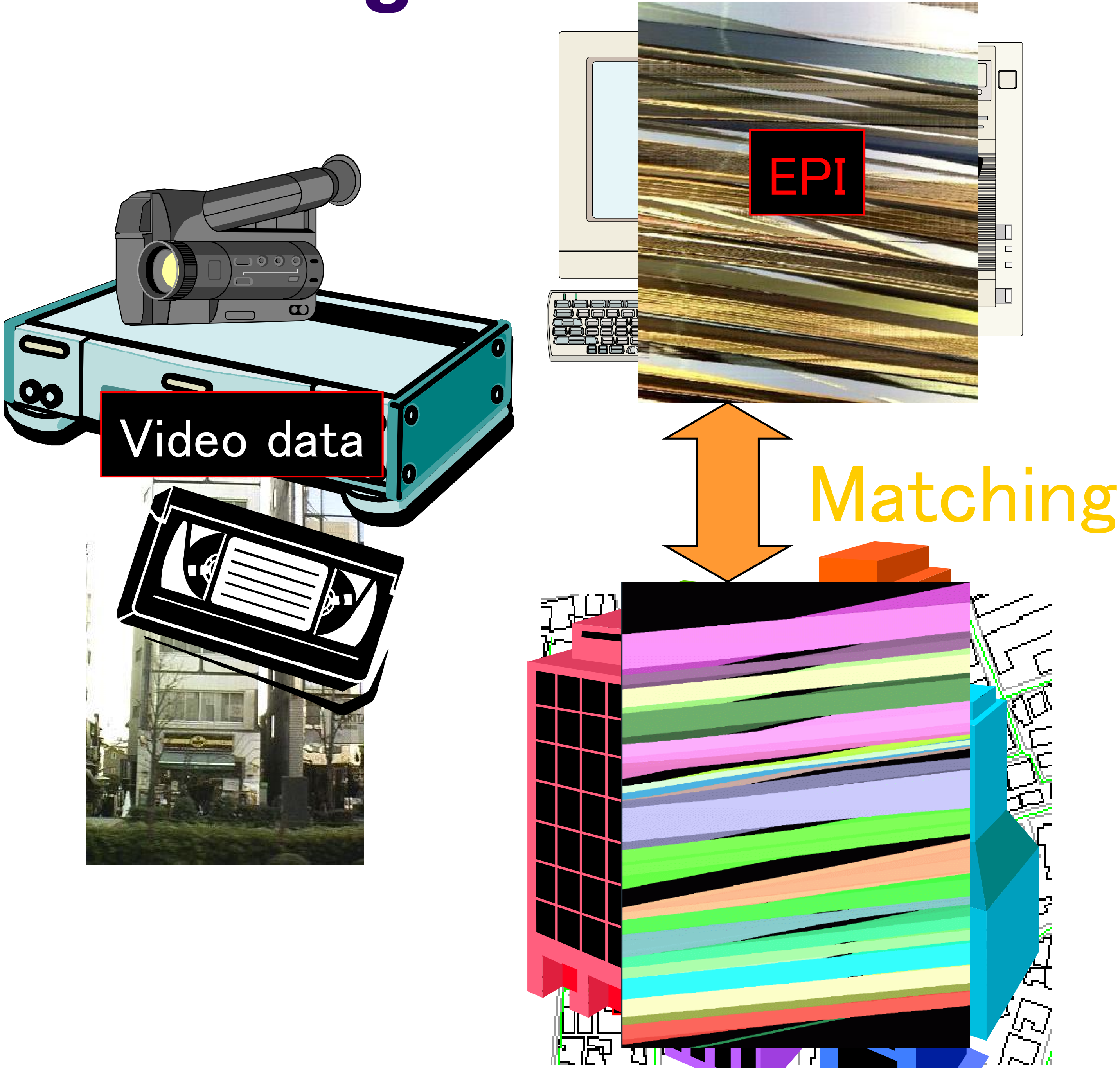
Depth Info

Digital residential map

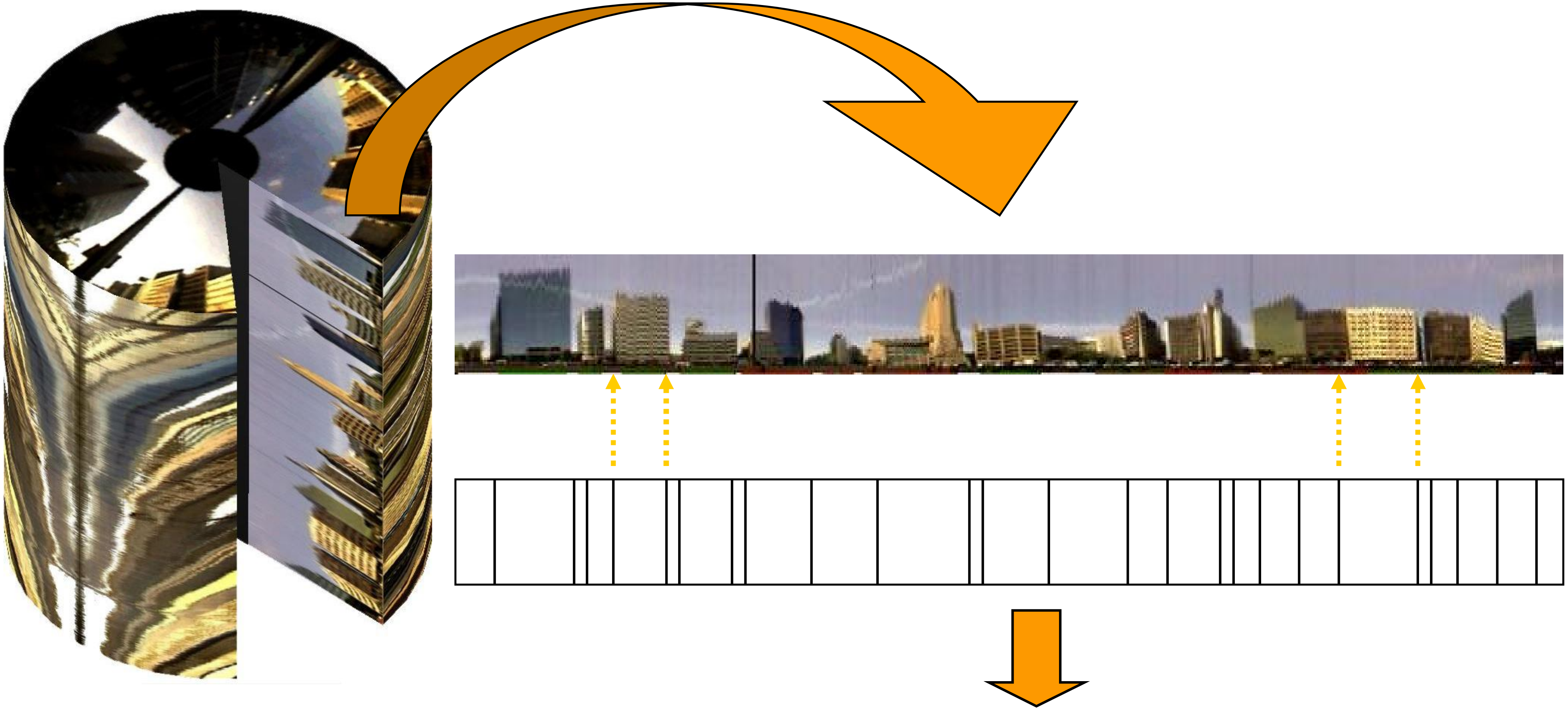


→ Correspondence between map and image

EPI Matching



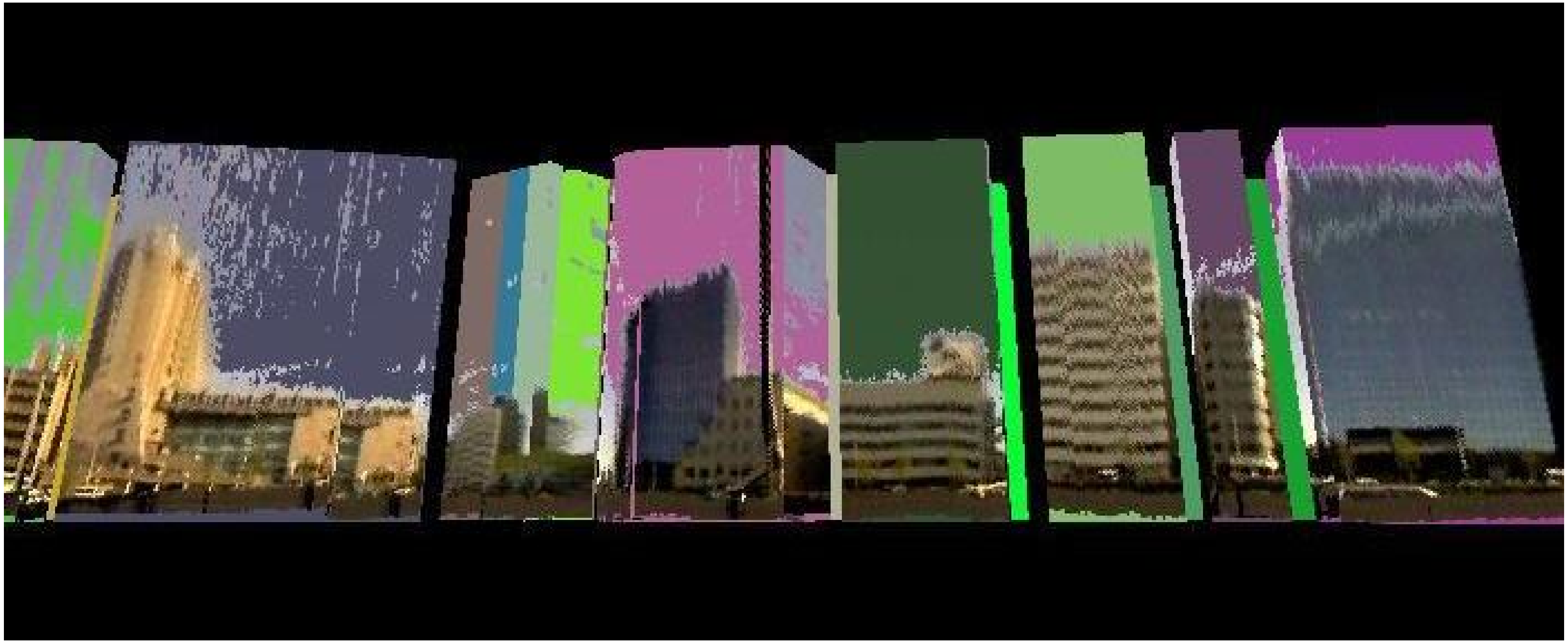
Cross-section (a radius line)



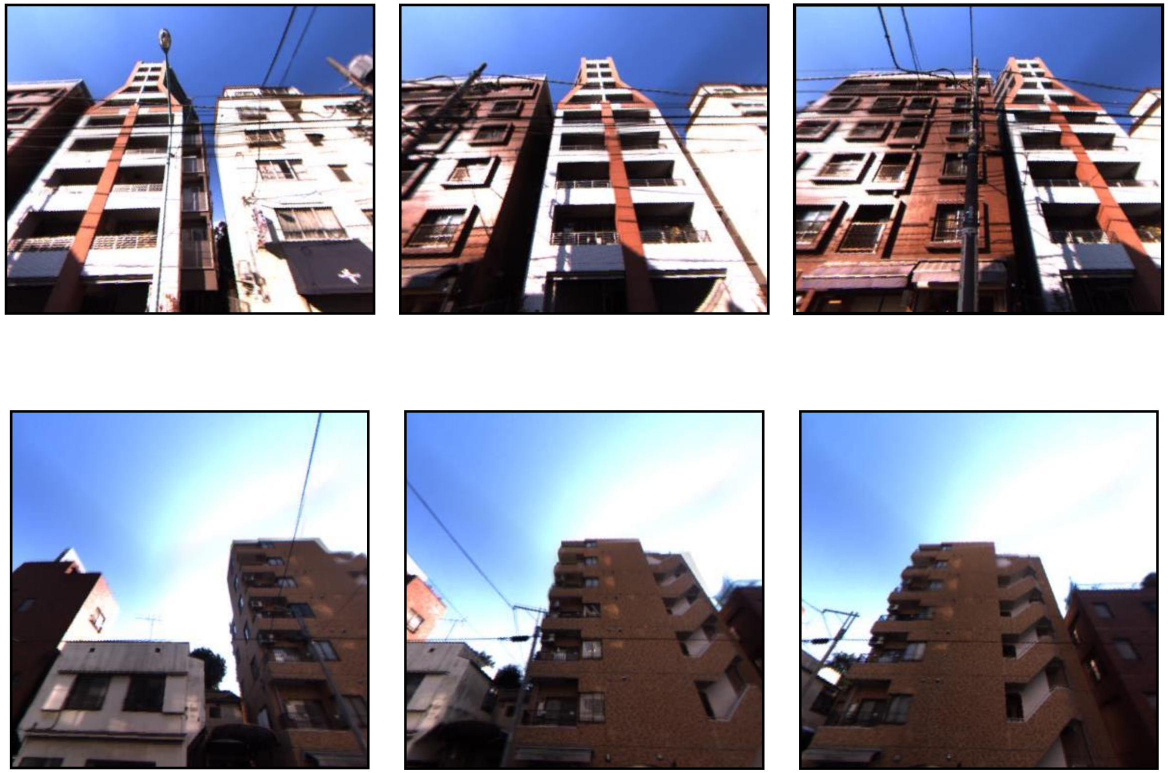
Panorama image

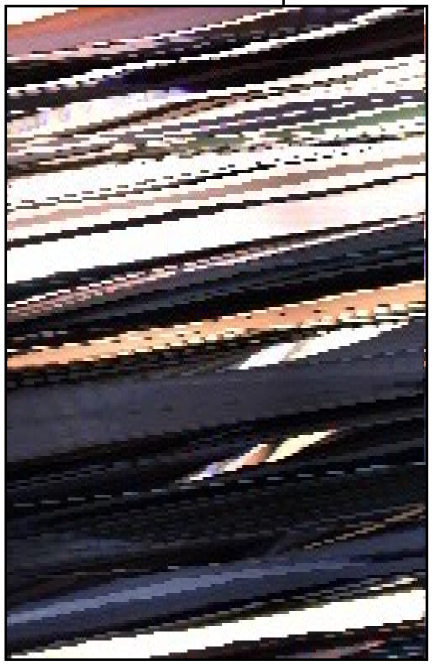
Texture Mapping

- Height info and texture



Problems in using EPI

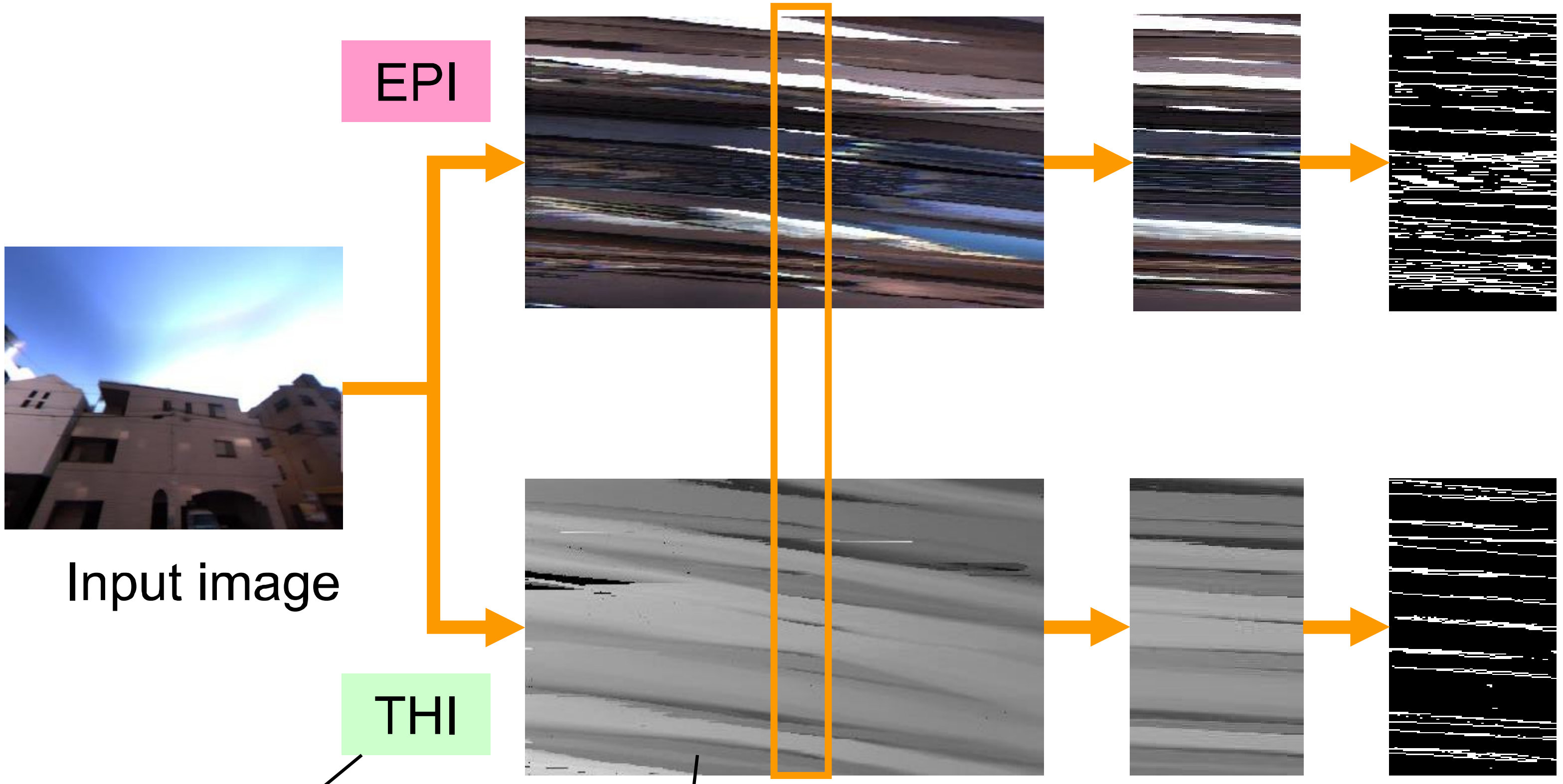




Real example
of EPI

Textures inside building (windows, etc.)
disturb to recognize the building features stably

Using Structural Information Instead of Color Information



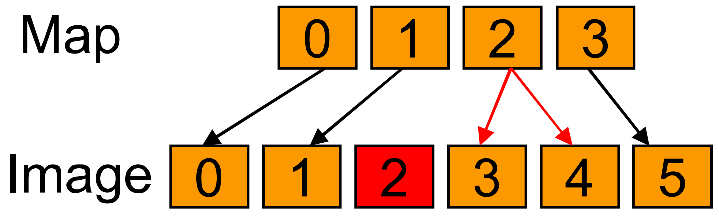
Input image

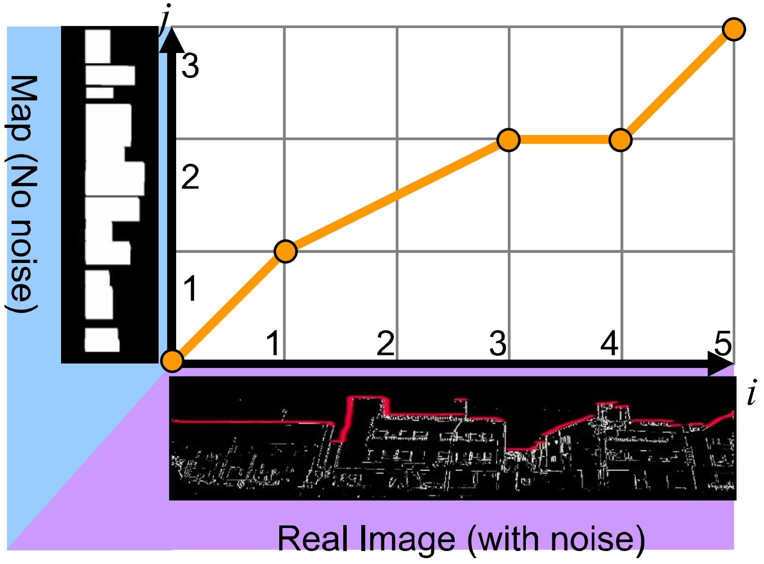
Temporal Height Image

Grey value ∝ Height (elevation angle) to the building roof

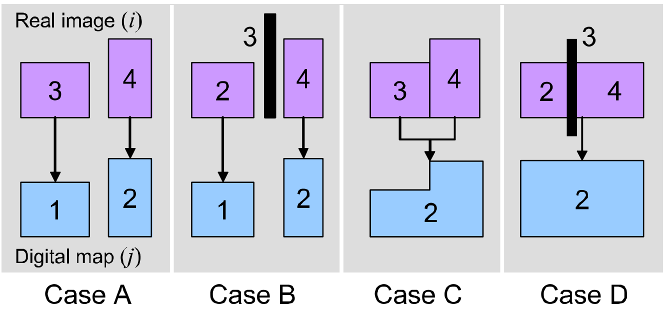
Much clearer building features

Building Matching between Map and Image using THI





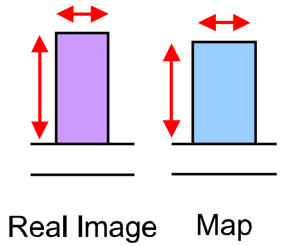
Matching Pattern



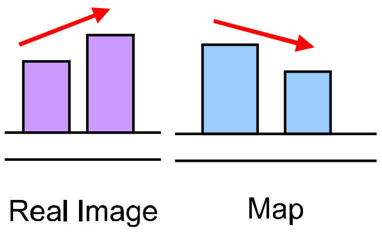
A: Corresponding 1 by 1
B: One noise between bldg.
C: Non-flat roof = two bldg.
D: One noise inside a bldg.

Matching Cost

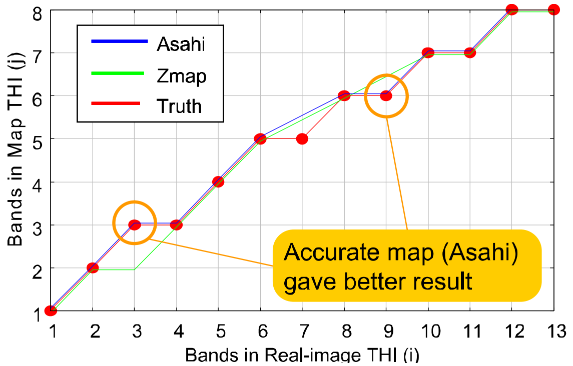
Aspect similarity



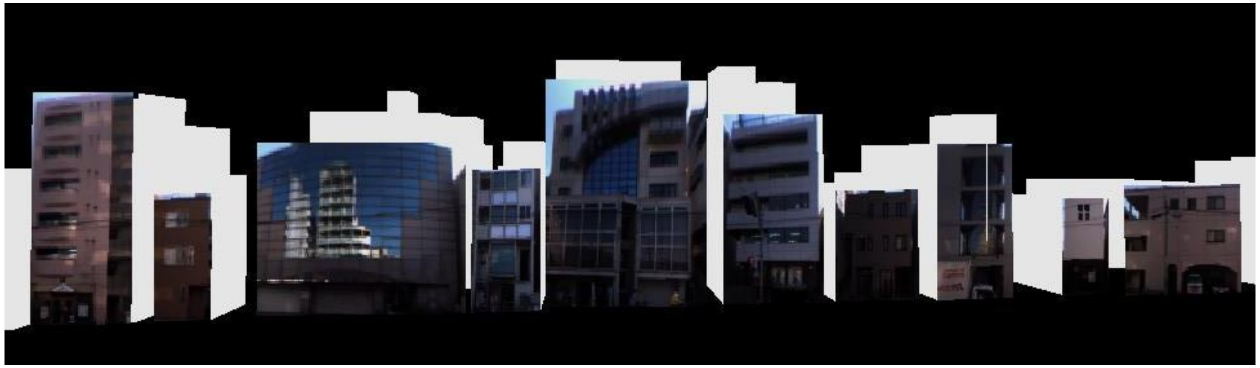
Height-transition similarity

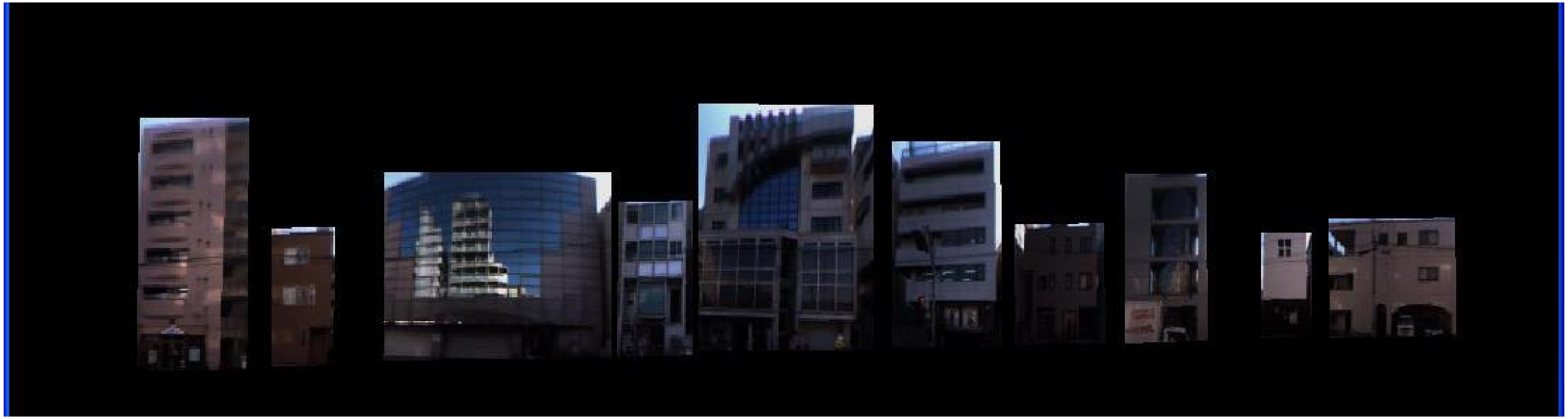


Matching and Texturing Result

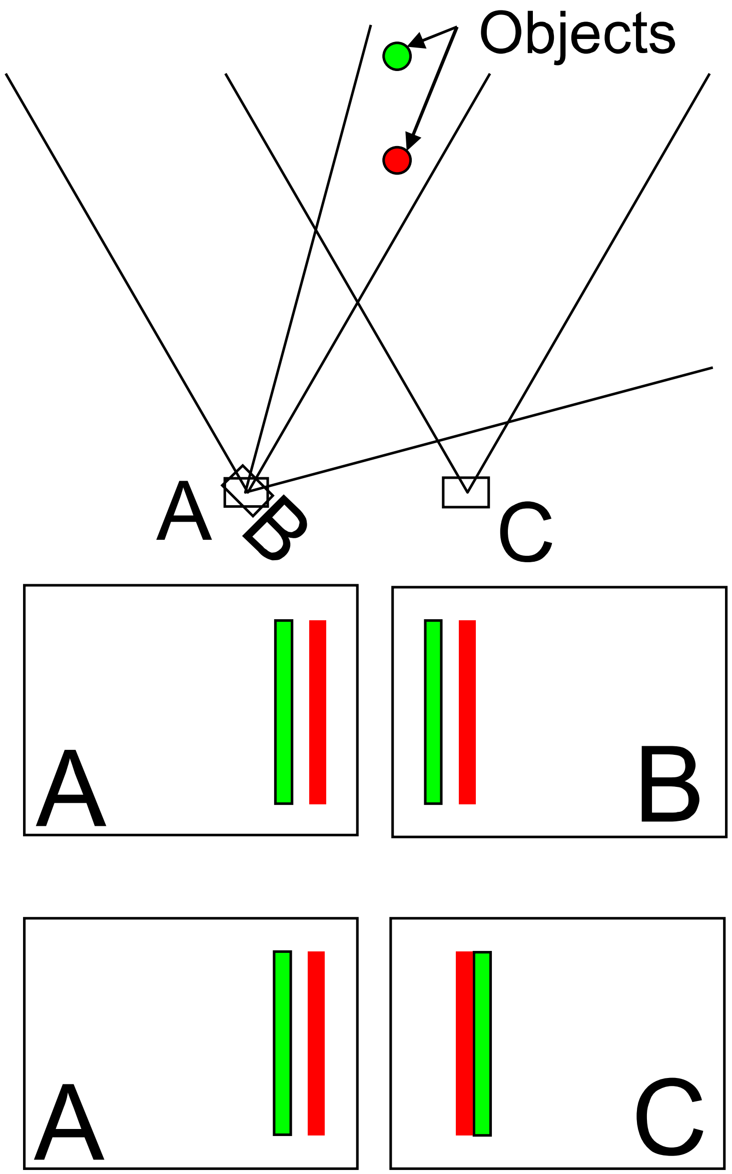


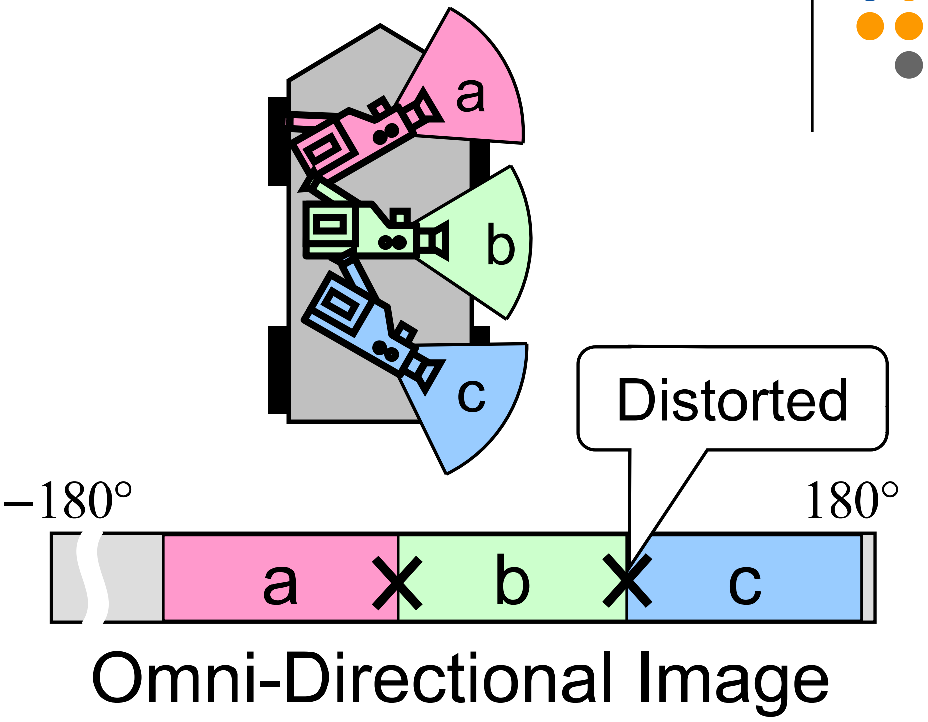
Accurate map (Asahi) gave better result

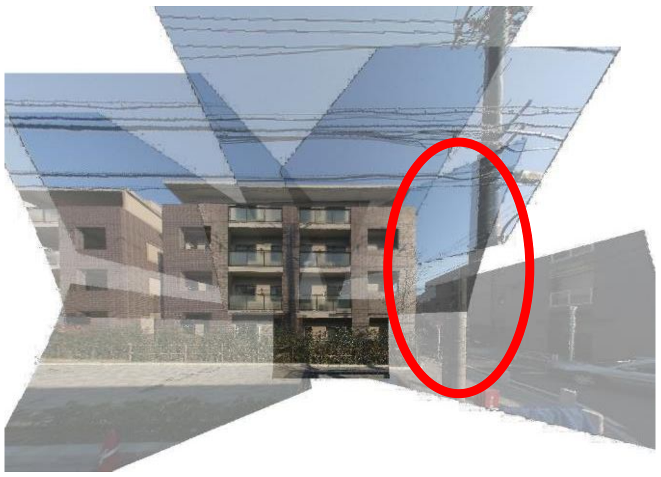




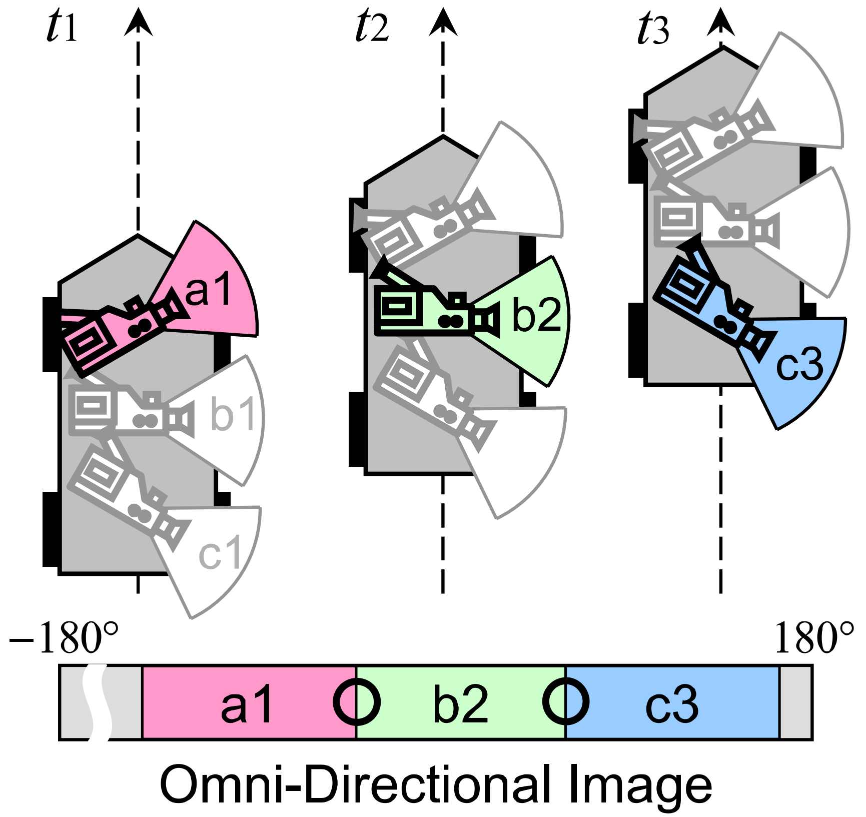
Camera center and distortion

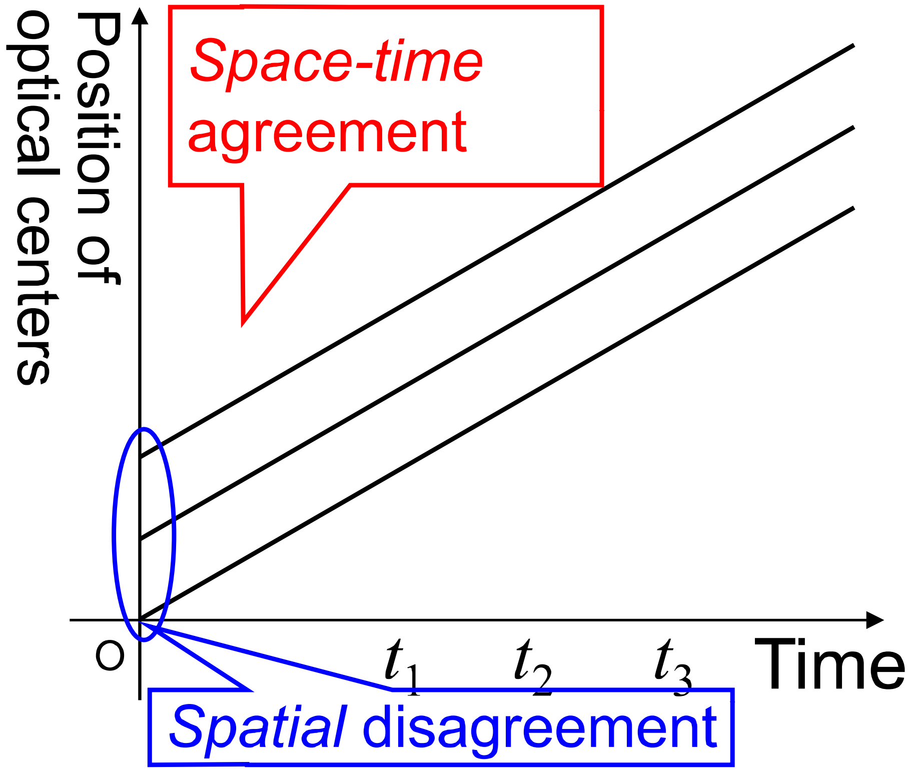






Spatio-temporal coincidence of camera optical center



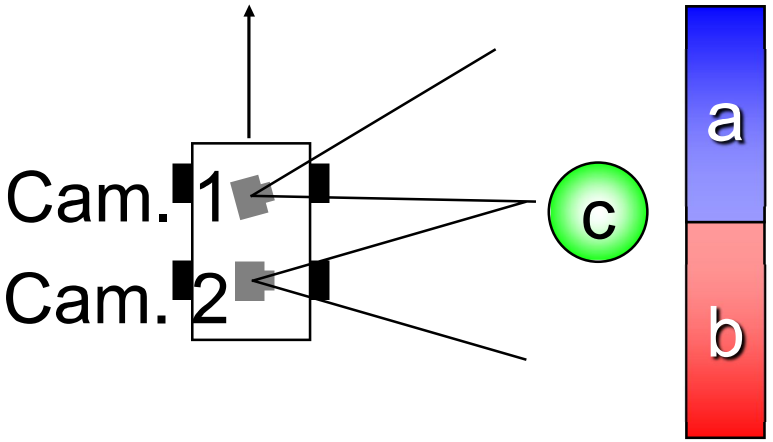


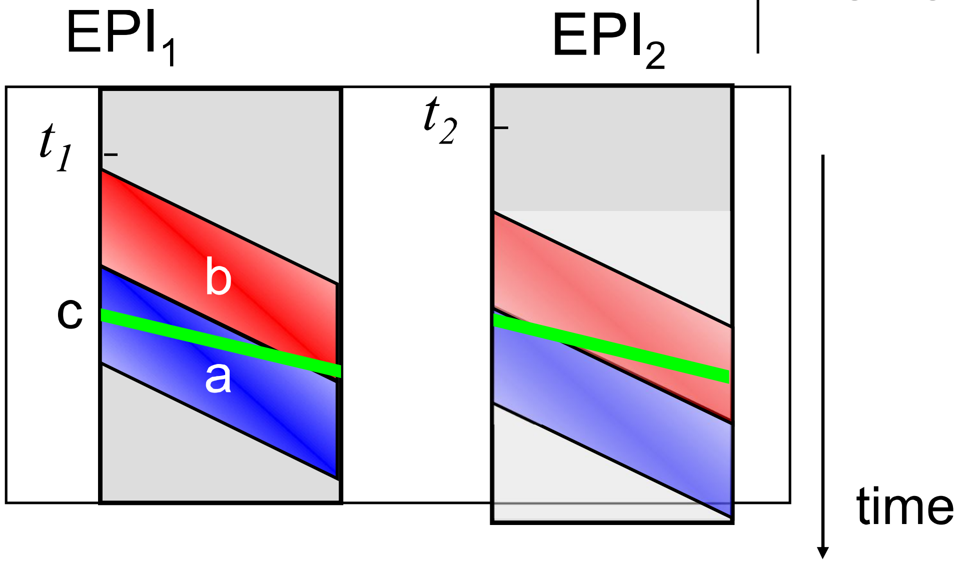
Space-time agreement

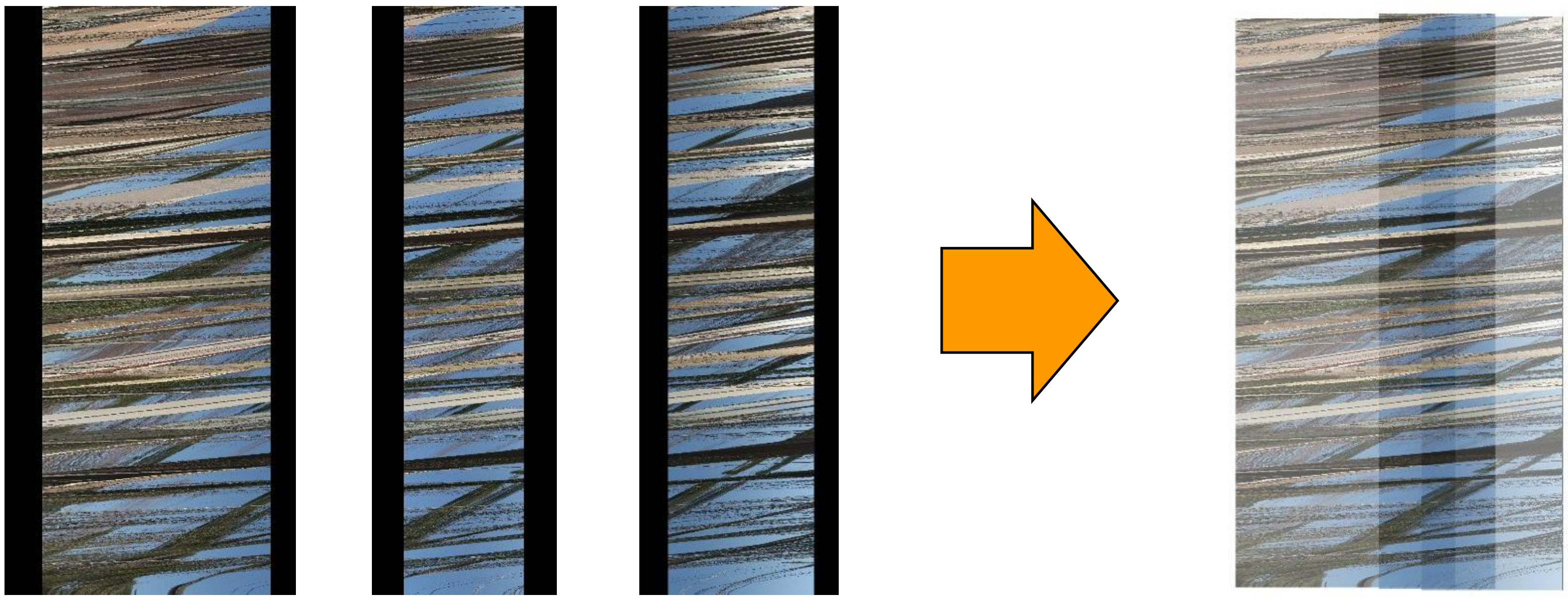
Spatial disagreement

How to know t2, t3?

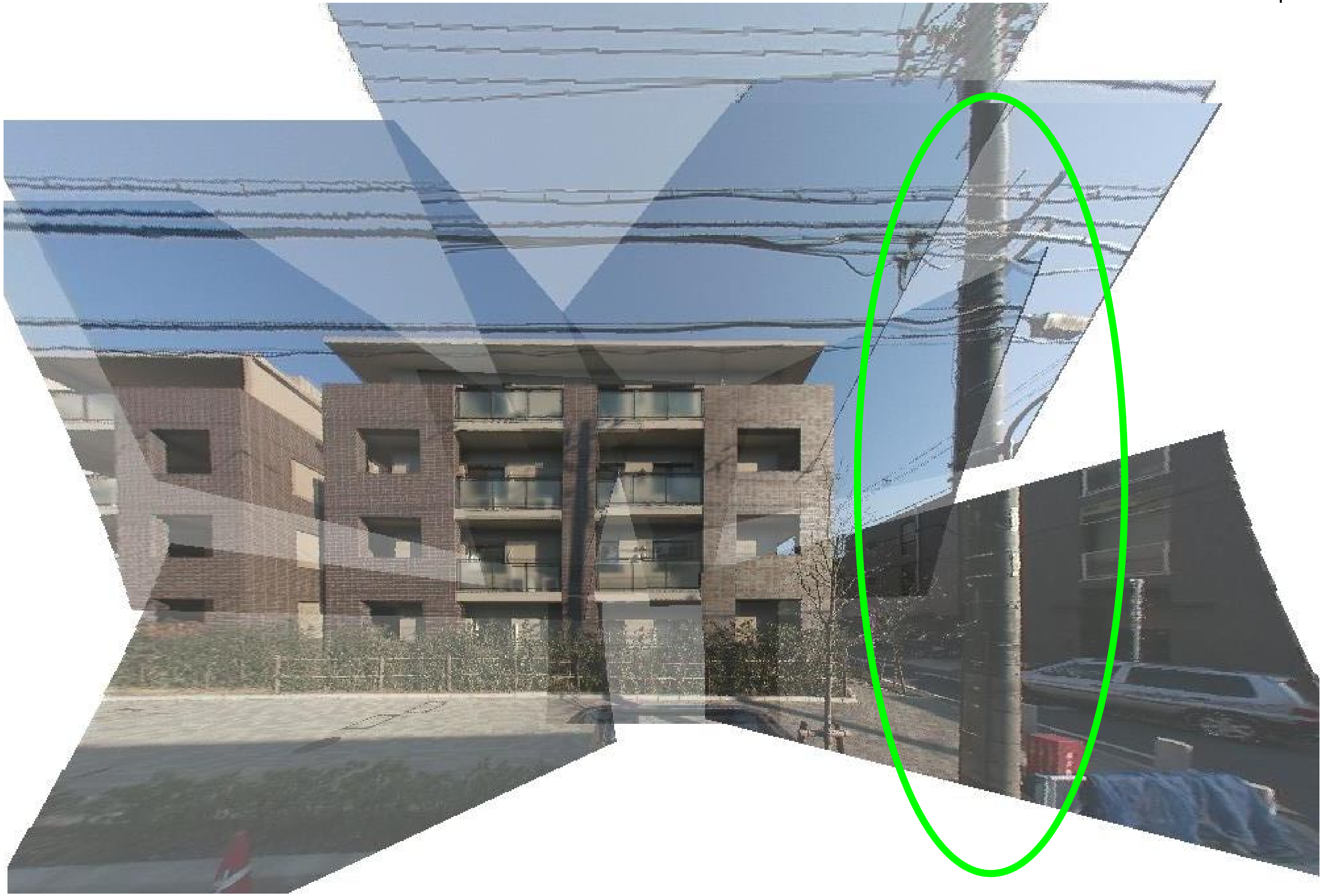
Temporal adjustment using EPI (Software-based camera sync.)





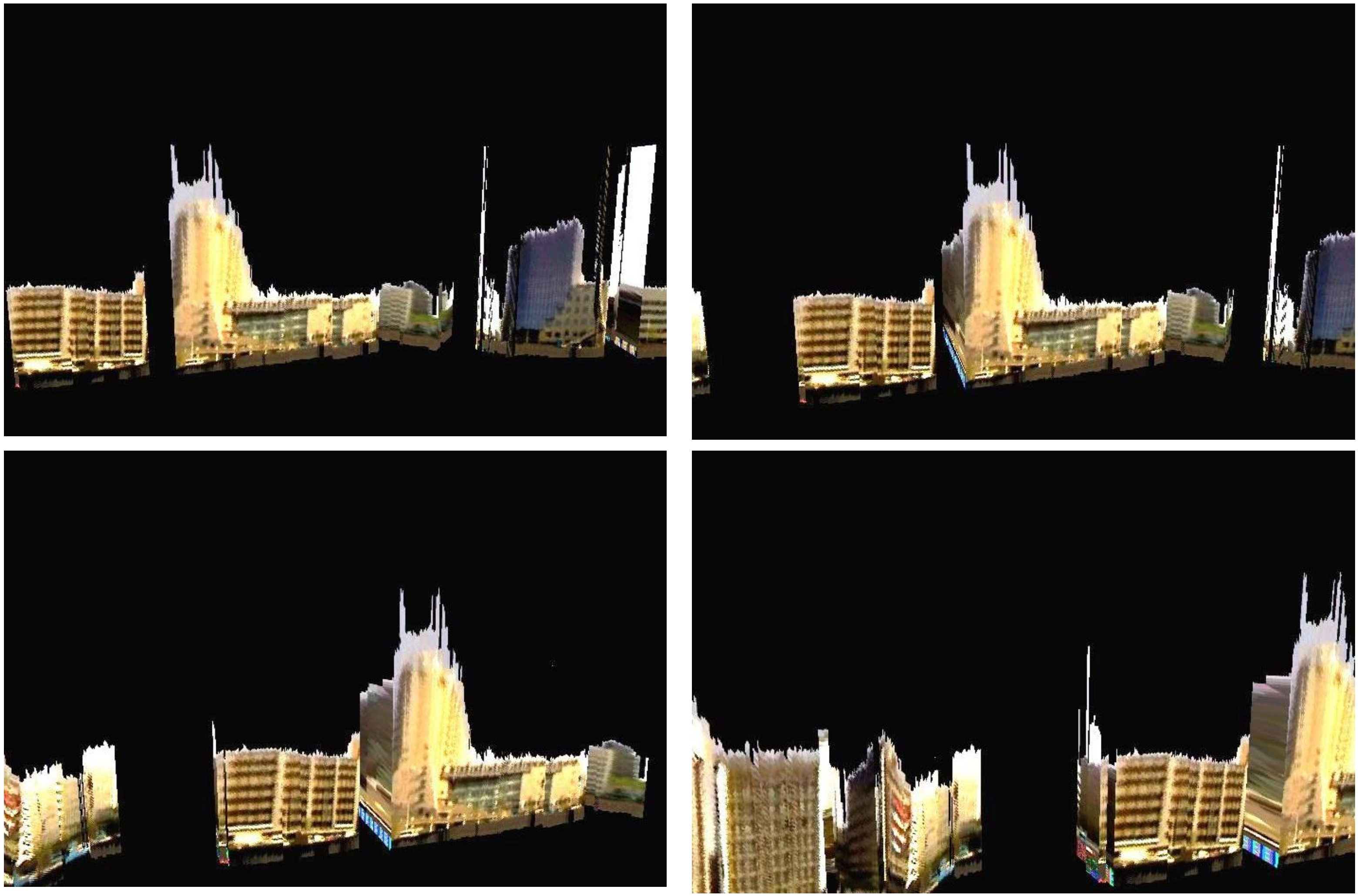


Result



Texture Mapping

- Side faces

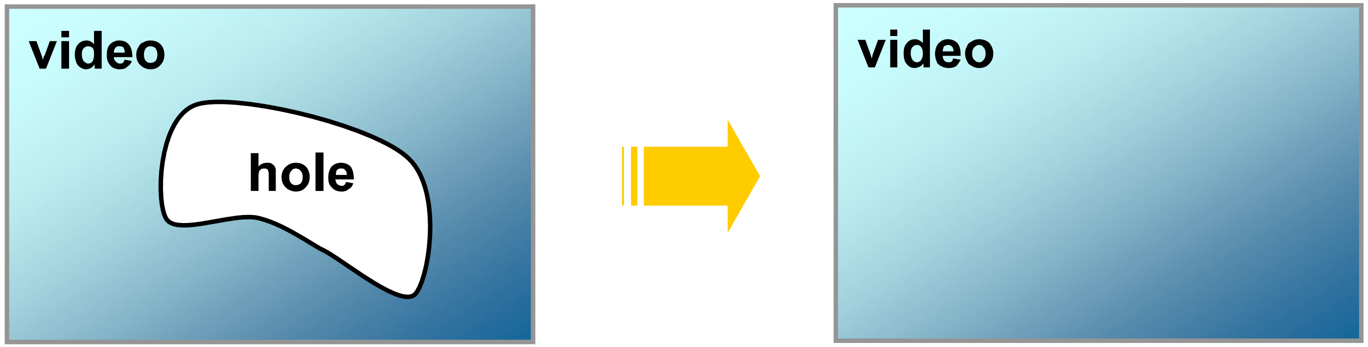


Video Completion

東京大学
THE UNIVERSITY OF TOKYO

Video Completion

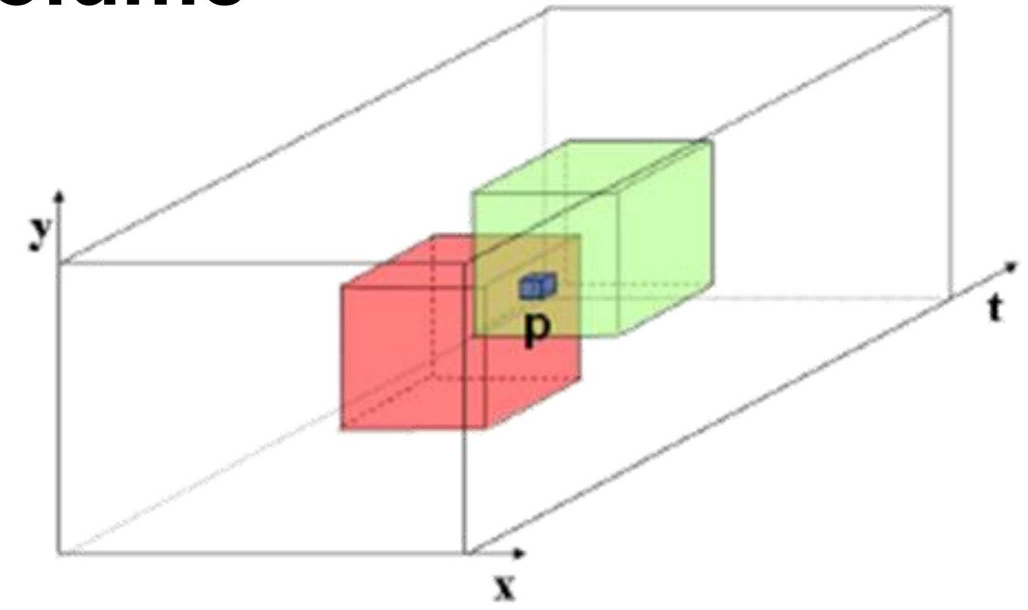
- **What's video completion?**



- **How is it useful?**
  - Restoration of damaged or vintage videos (Spatial completion)
  - Restoration of corrupted internet video streams due to packet drops (Temporal completion)
  - Post-production in the movie-making industry

Space-time Completion of Video
(Y. Wexler* 2004, 2007)

- Find a small volume which accords with the hole from the whole volume
- Copy it to the hole, as a compensating patch-volume



How to Find the Patch

The optimal patch-volume

Reference database =The whole volume itself

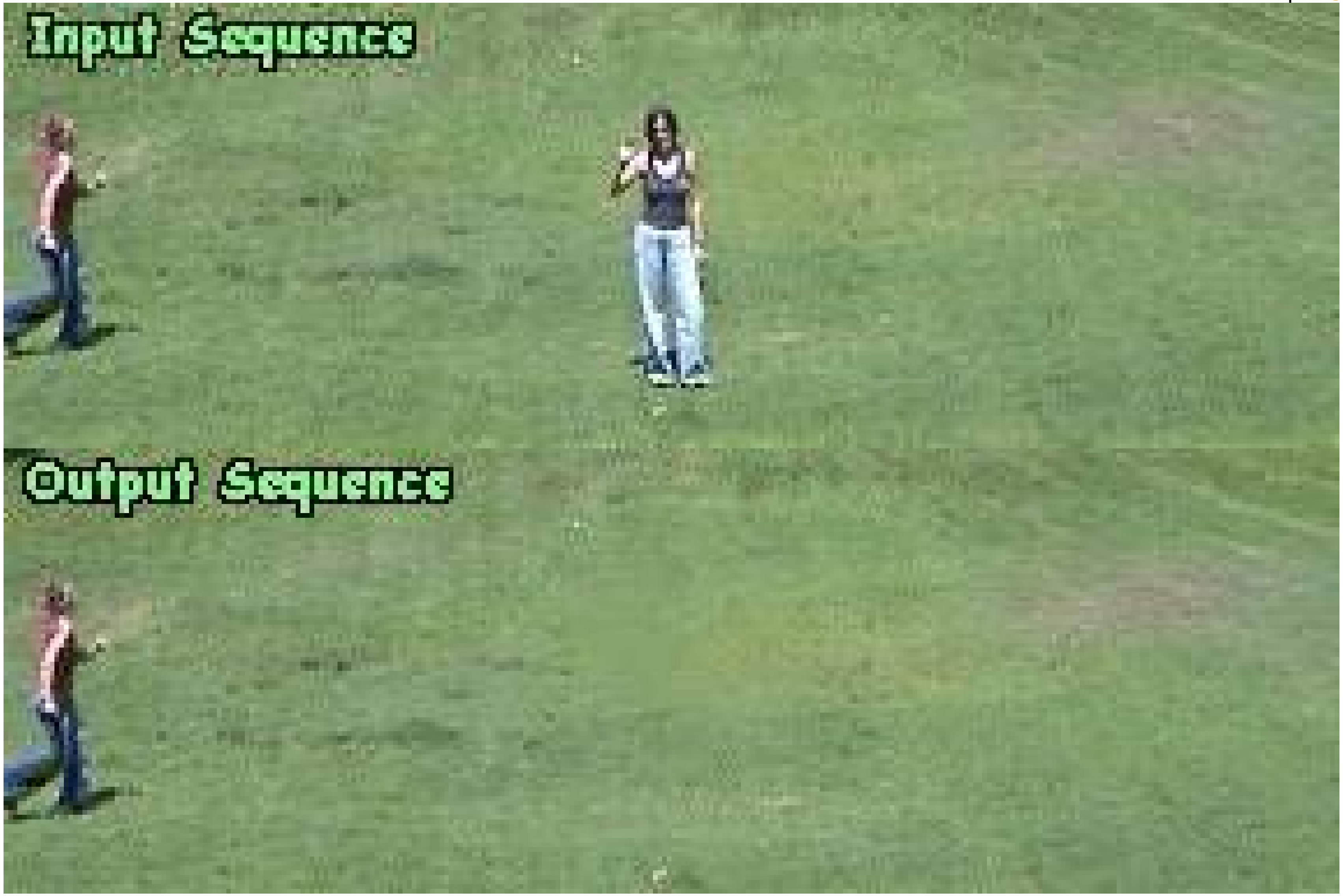
A small cube around point *p*

$$\mathrm{Coherence}(\mathcal{S}^*|\mathcal{T}) = \sum_{p\in\mathcal{S}^*} \max_{q\in\mathcal{T}} s\left(W_p, W_q\right)$$

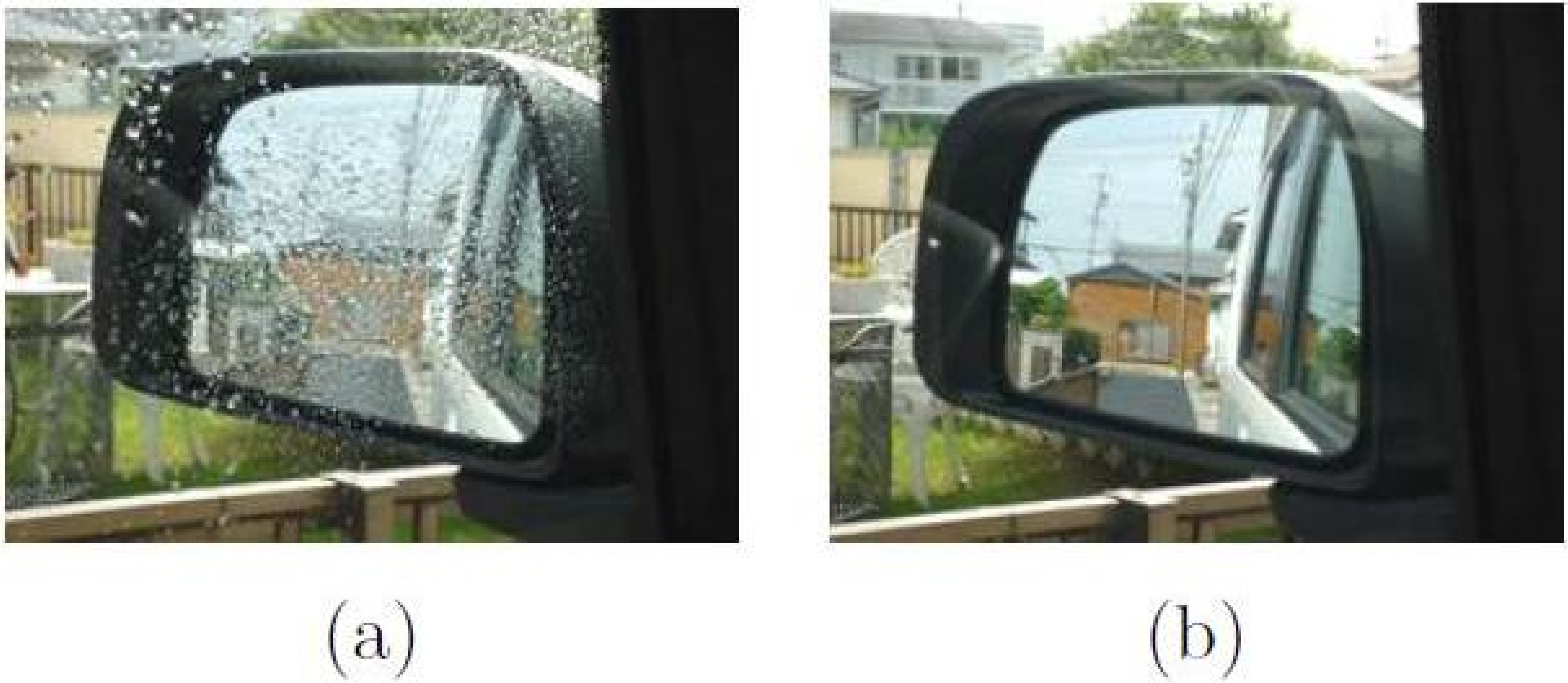
$$s(W_p, W_q) = e^{\frac{-d(W_p,W_q)}{2*\sigma^2}}$$

$$d(W_p, W_q) = \sum_{(x,y,t)} ||W_p(x,y,t) \dot{-} W_q(x,y,t)||^2$$

Result



Erasing Raindrop
[J. Sato et al. 2011]



(a)

(b)

Fig.1 雨滴付き画像と雨滴なし画像.

Erasing Raindrop
[J. Sato et al. 2011]

Key Point:
The camera is mounted on a vehicle
Always fixed to observe the mirror

We can know from which portion of ST-volume the raindrop can be inpainted.

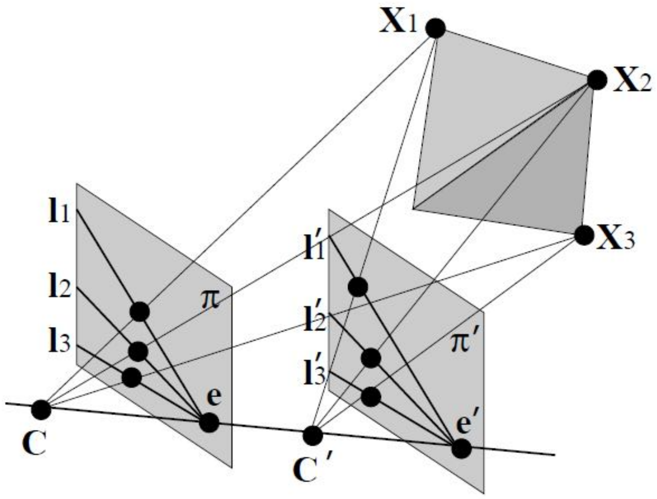


Fig.2 並進カメラの自己エピポーラ幾何

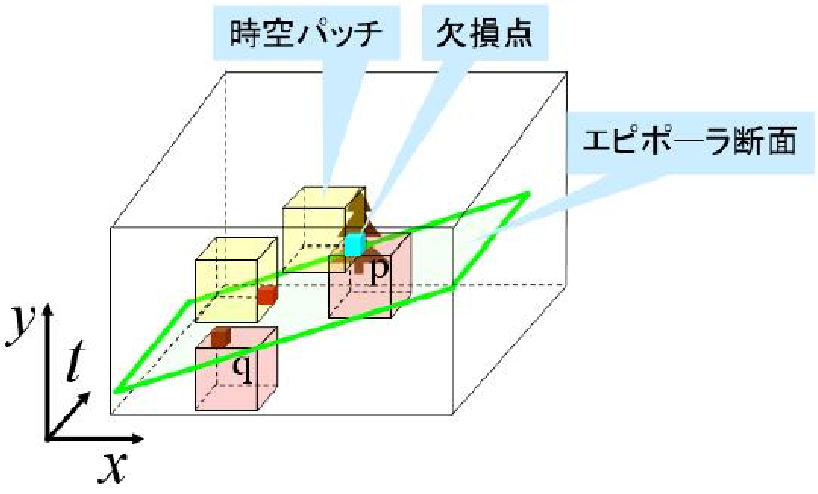
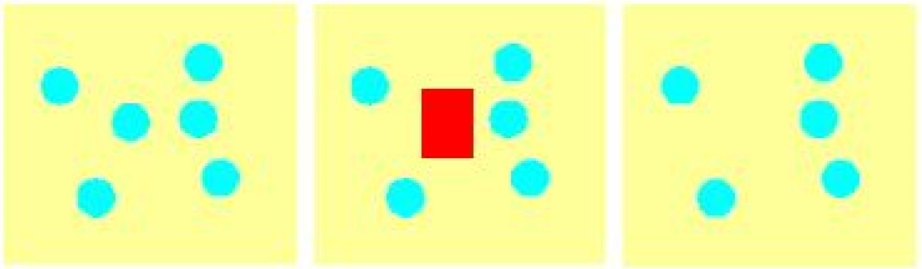


Fig.4 欠損点の補間のための時空パッチ

Epipolar Geometry:
(Details in *“Stereo Vision”*, Nov./Dec.)

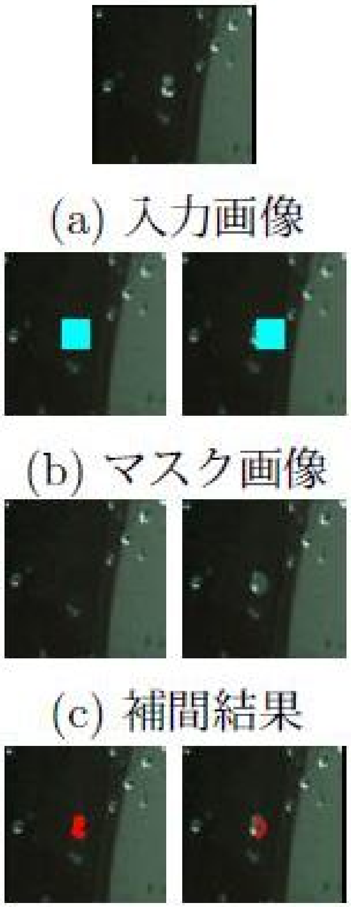
Detecting the Raindrop

1. Restore the masked area
2. Restore the whole image by shifting the mask
3. Subtract the restored image from the original image



(a) 入力画像　(b) マスク画像　(c) 補間画像

Fig.5 画像補間を用いた雨滴検出



(a) 入力画像

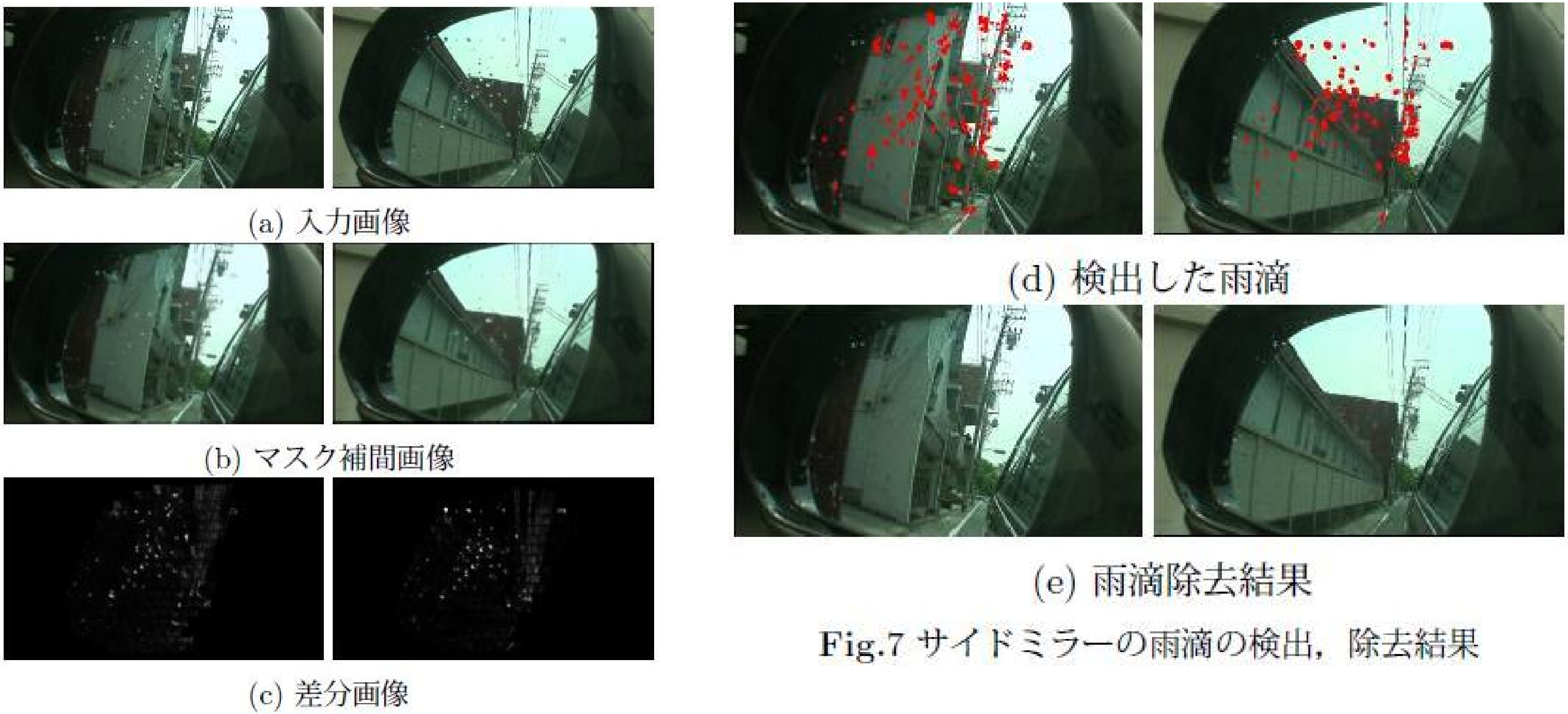
(b) マスク画像

(c) 補間結果

(d) 差分からの雨滴検出結果

Fig.6 マスク画像の補間を利用した雨滴検出

Result



(a) 入力画像

(b) マスク補間画像

(c) 差分画像

(d) 検出した雨滴

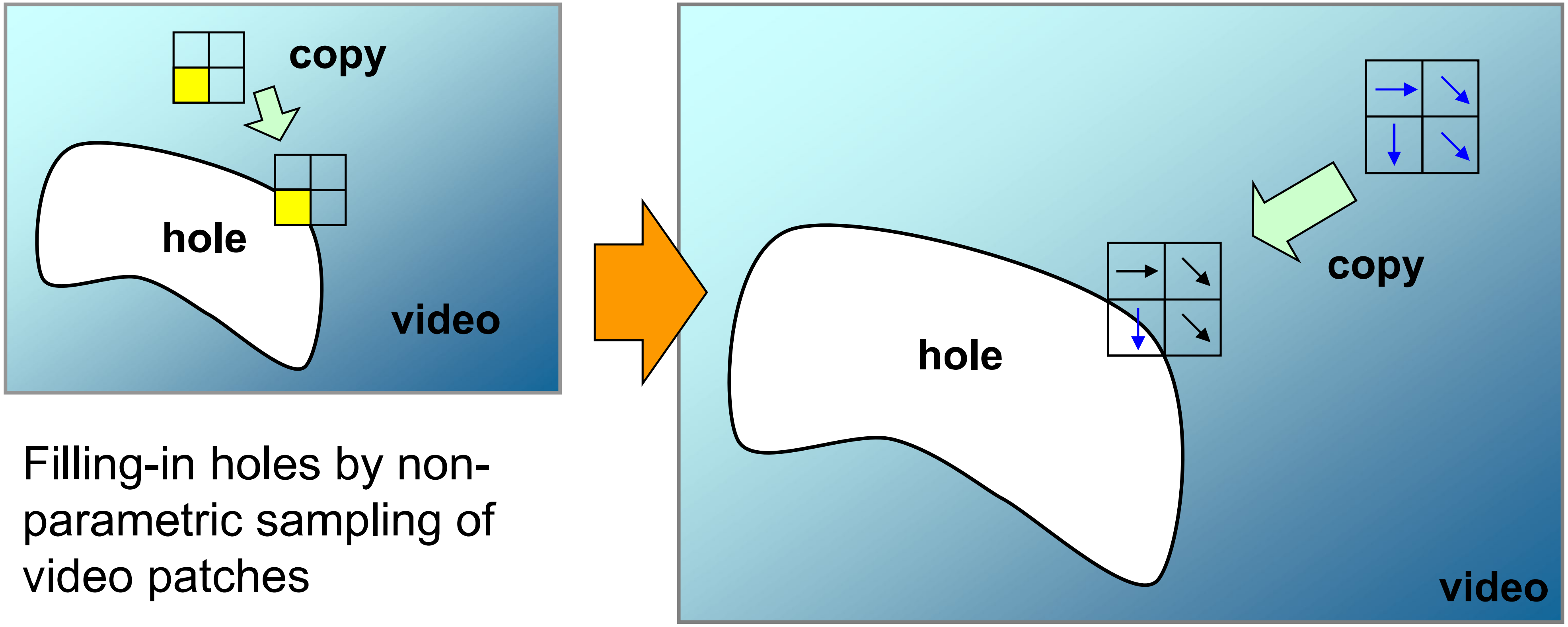
(e) 雨滴除去結果

Fig.7 サイドミラーの雨滴の検出，除去結果

Video Completion by Motion Field Transfer (Shiratori 2006)

Conventional

Proposed

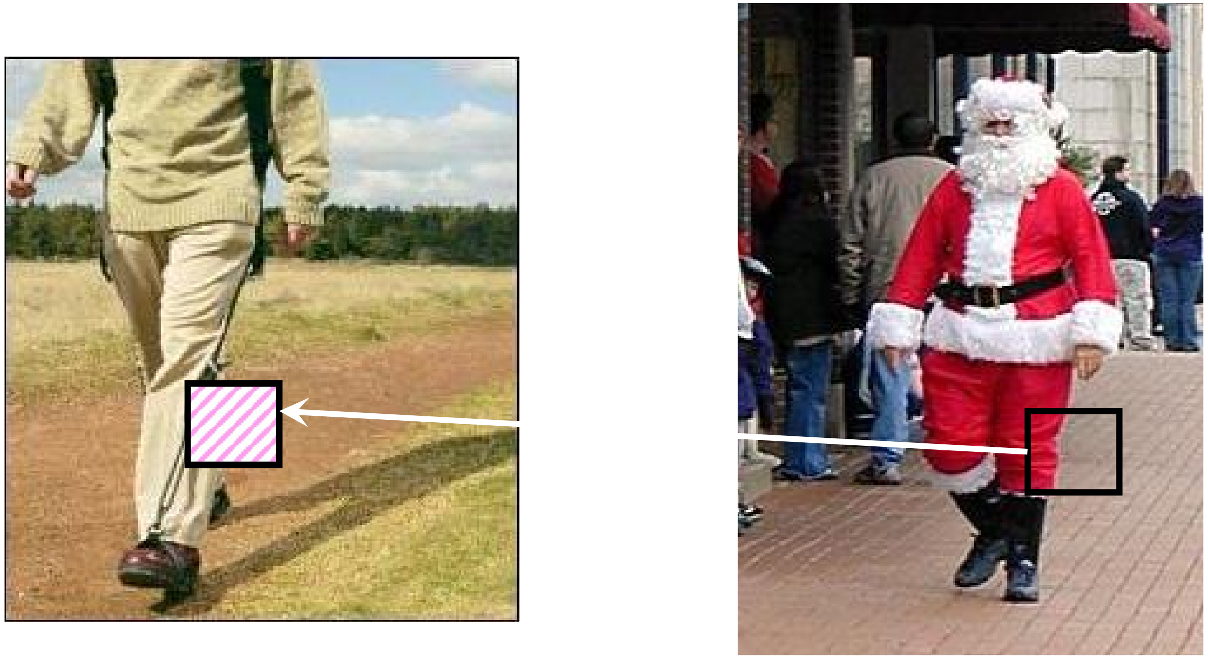


Filling-in holes by non-parametric sampling of video patches

Why motion?

- Color-based method : Requires similar <u>color</u> & <u>motion</u>
- Motion-based method : Requires only similar <u>motion</u>

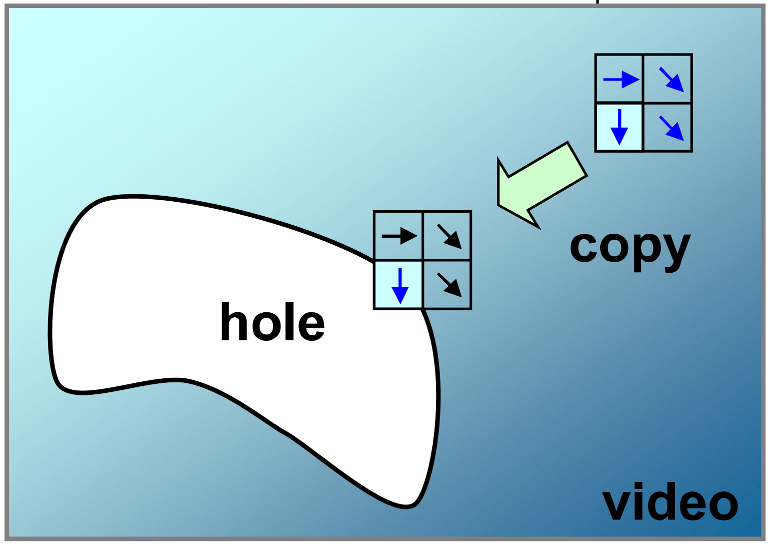
➡ More chance to fill-in a hole!



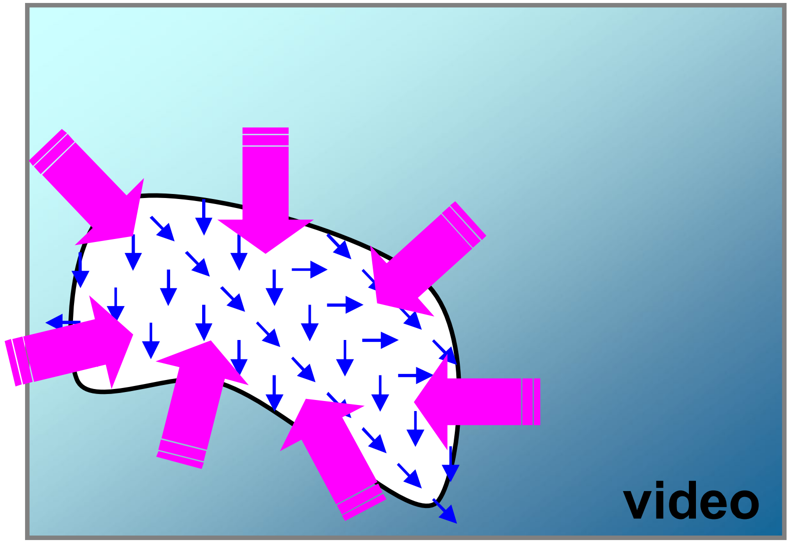
Motion can be copied from video portions with *different appearance.*

Method

- **Motion Field Transfer**
  - Fill-in a hole by transferring the most similar *motion patches*



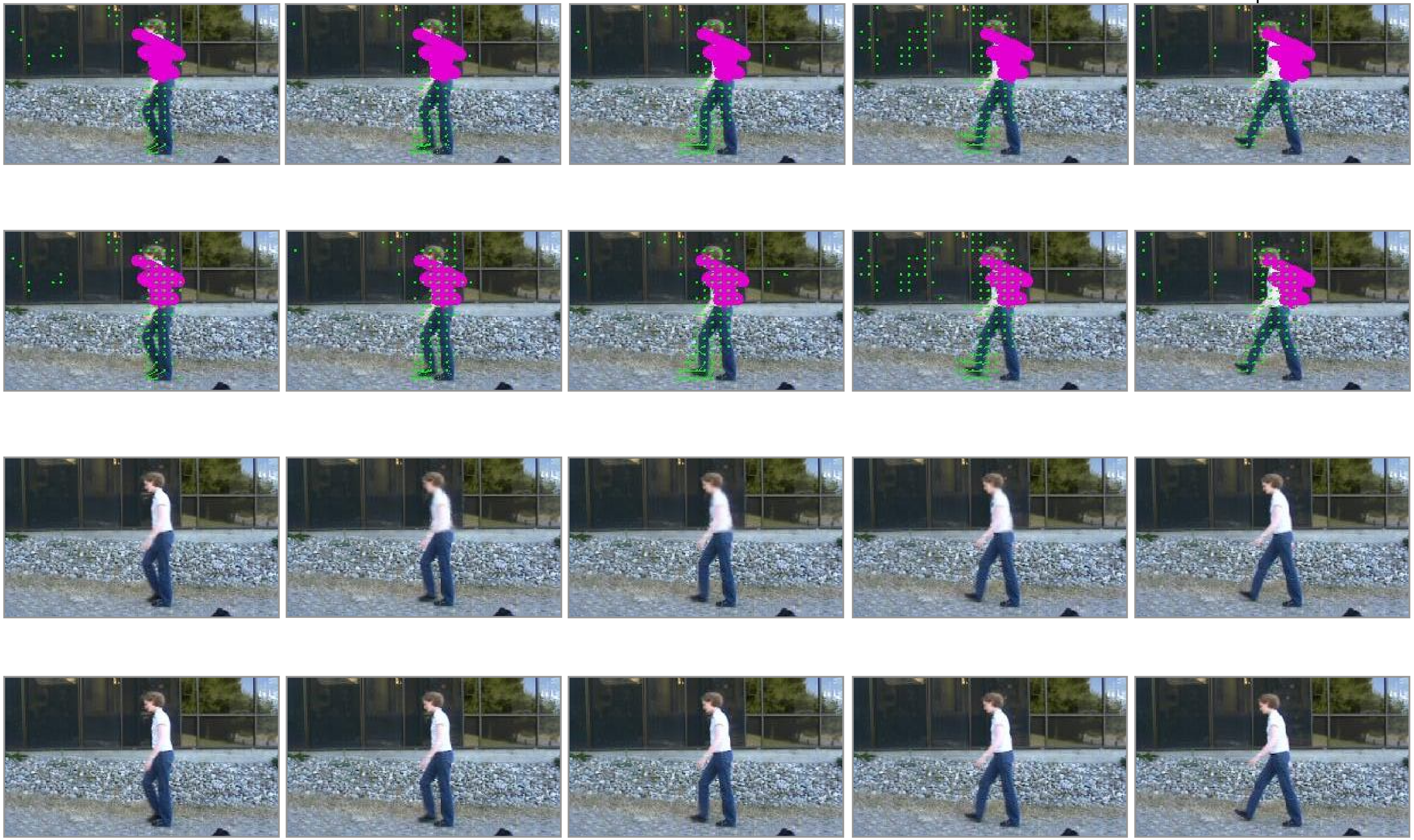
- **Color Propagation**
  - Propagate color from boundary using motion field in the hole



Result

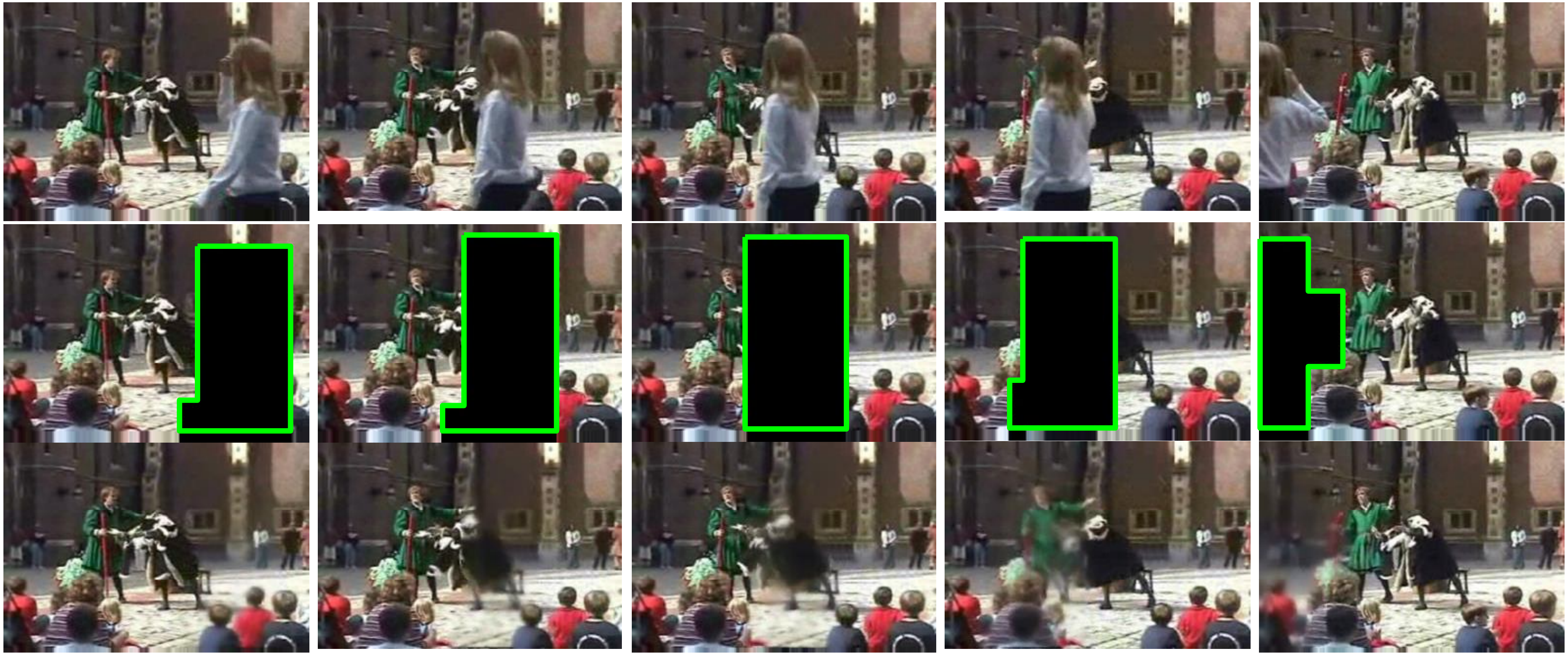
[redacted]

Result



Application: Object Removal

Application: Object Removal



Application: Frame Interpolation

Application: Frame Interpolation(2)

Full-Frame Video Stabilization
(Y. Matsushita 2006)

- **Motion inpainting (propagating local motion into the missing image areas)**

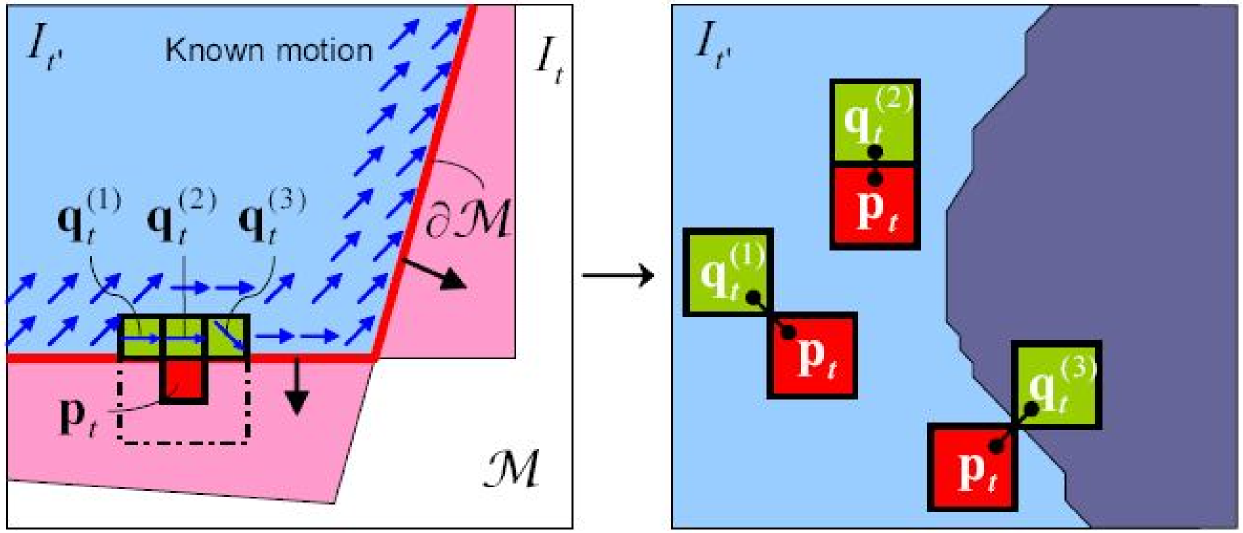
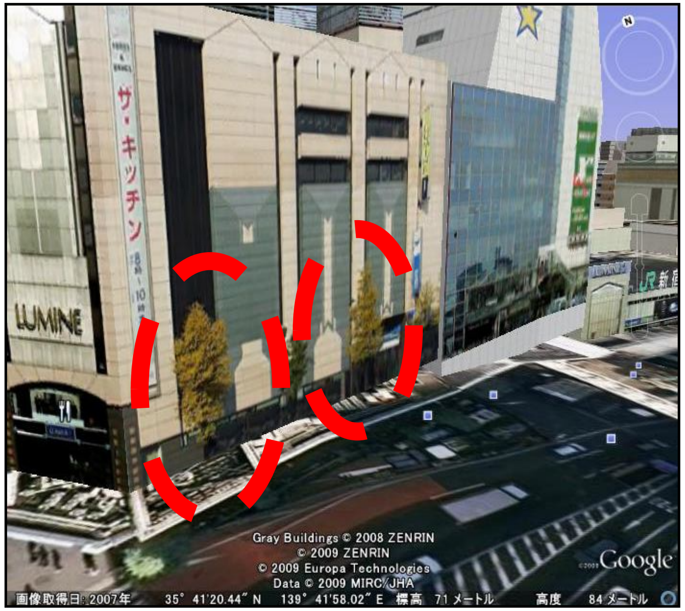


Figure 5: *Motion inpainting. Motion field is propagated on the advancing front $\partial\mathcal{M}$ into $\mathcal{M}$. The color similarities between $\mathbf{p}_t$ and its neighbors $\mathbf{q}_t$ are measured in the neighboring frame $I_{t'}$ after warped by local motion of $\mathbf{q}_t$, and they are used as weight factors for the motion interpolation.*

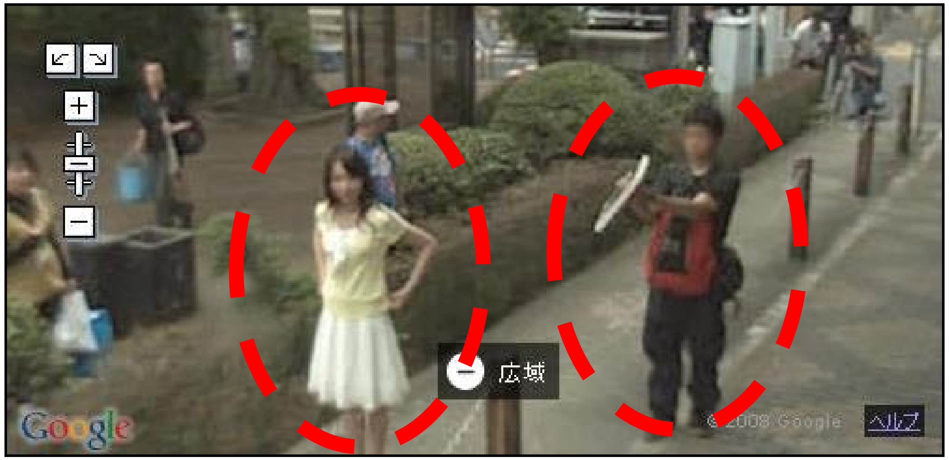
Result

Removing Foreground Objects
[Kuribayashi 2009]

Wrong texture mapping
Pedestrian's privacy


LUMINE
ザ・キッチン
Gray Buildings © 2008 ZENRIN
© 2009 ZENRIN
© 2009 Europa Technologies
Data © 2009 MIRC/JHA
Google
画像取得日 2007年 35° 41'20.44" N 139° 41'58.02" E 標高 71 メートル 高度 84 メートル

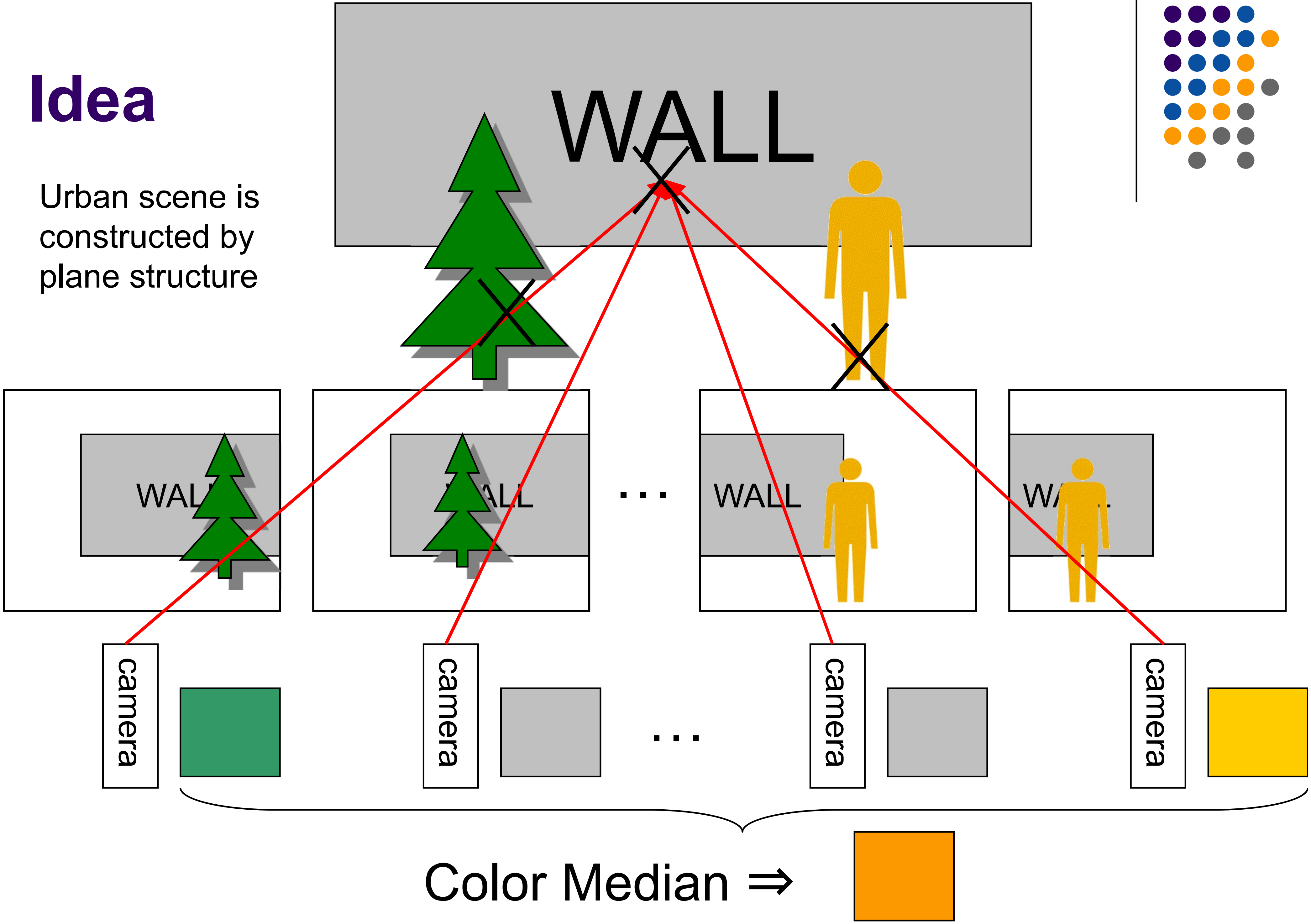
Google Earth



Google Street View

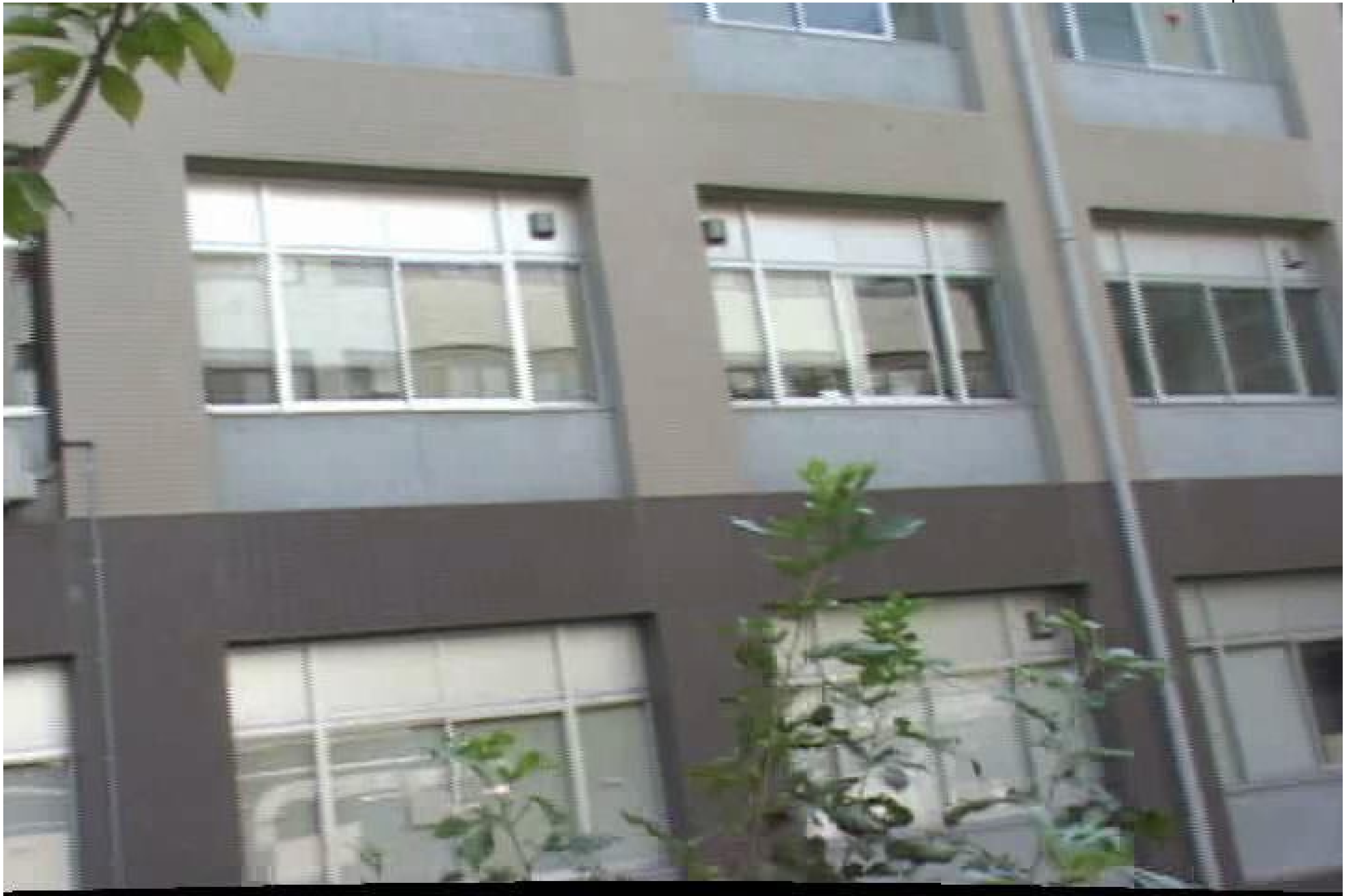
Idea

Urban scene is constructed by plane structure



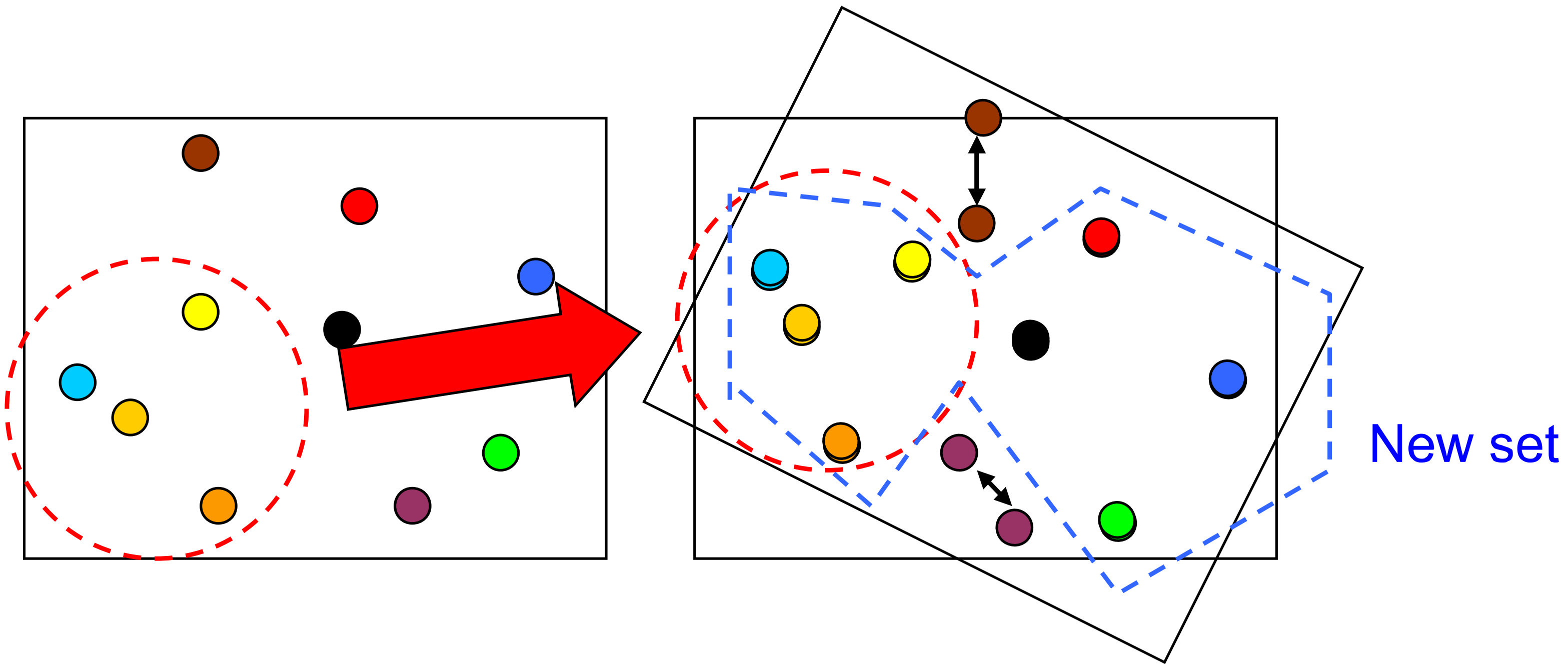
Color Median ⇒

Input



Plane-Plane Registration

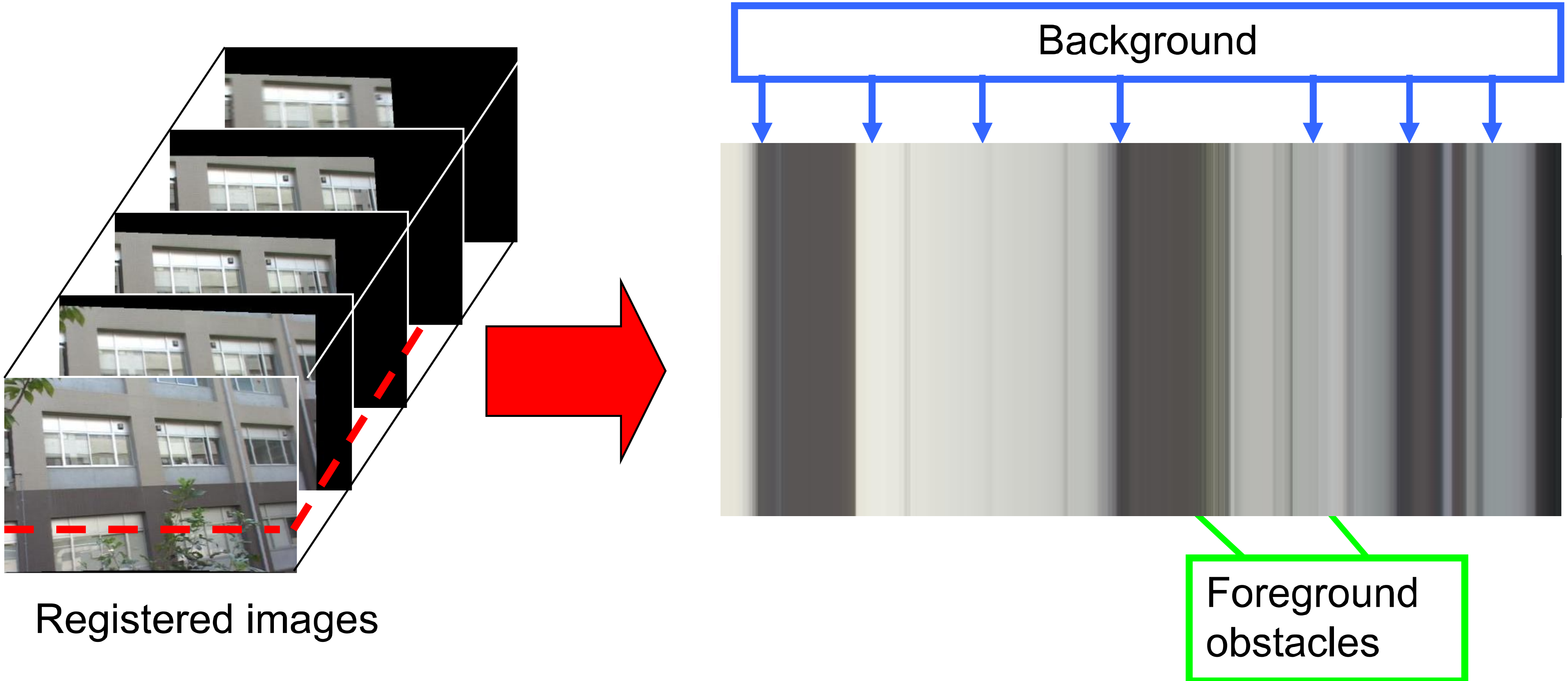
- SIFT + Homography + RANSAC



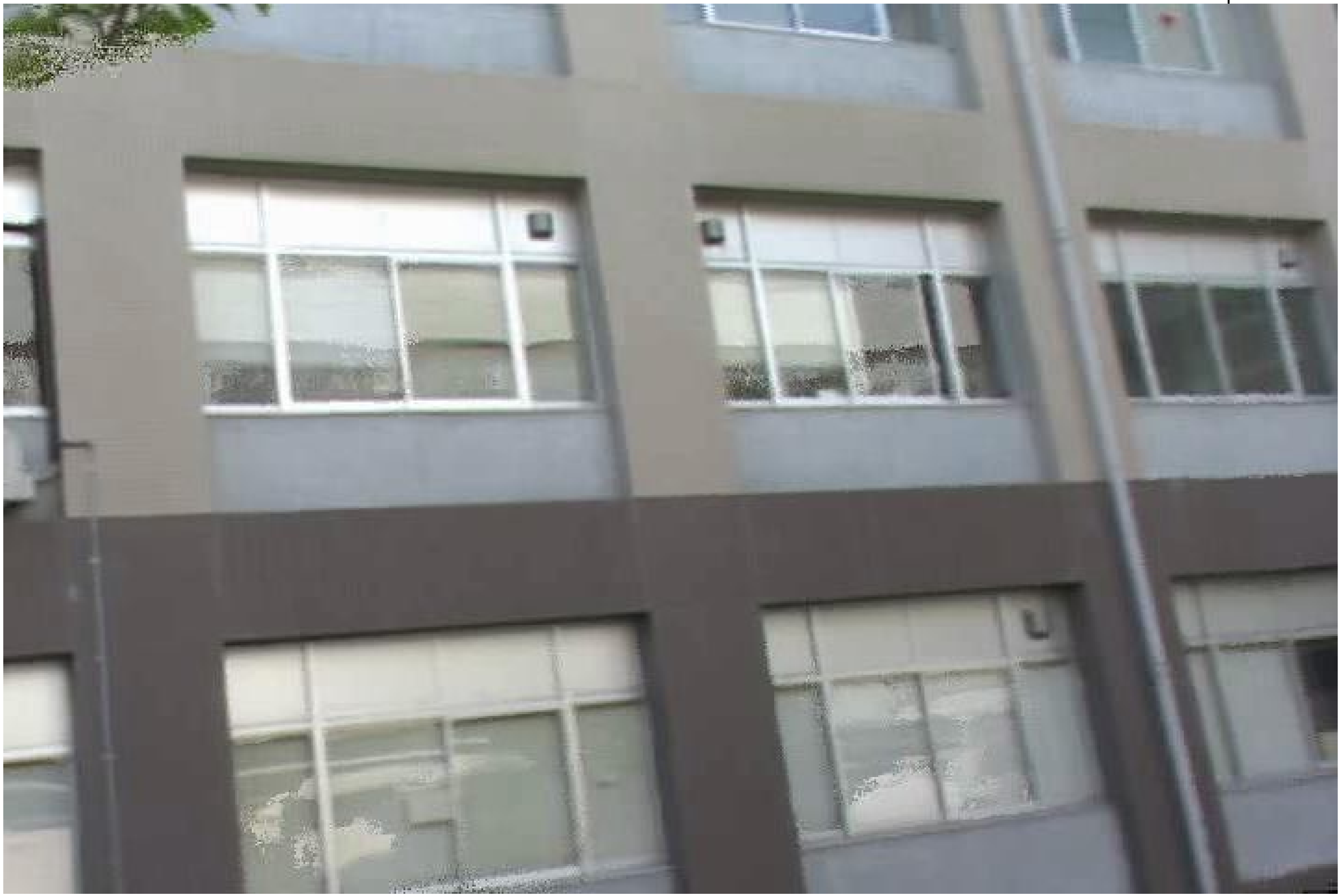
Registered

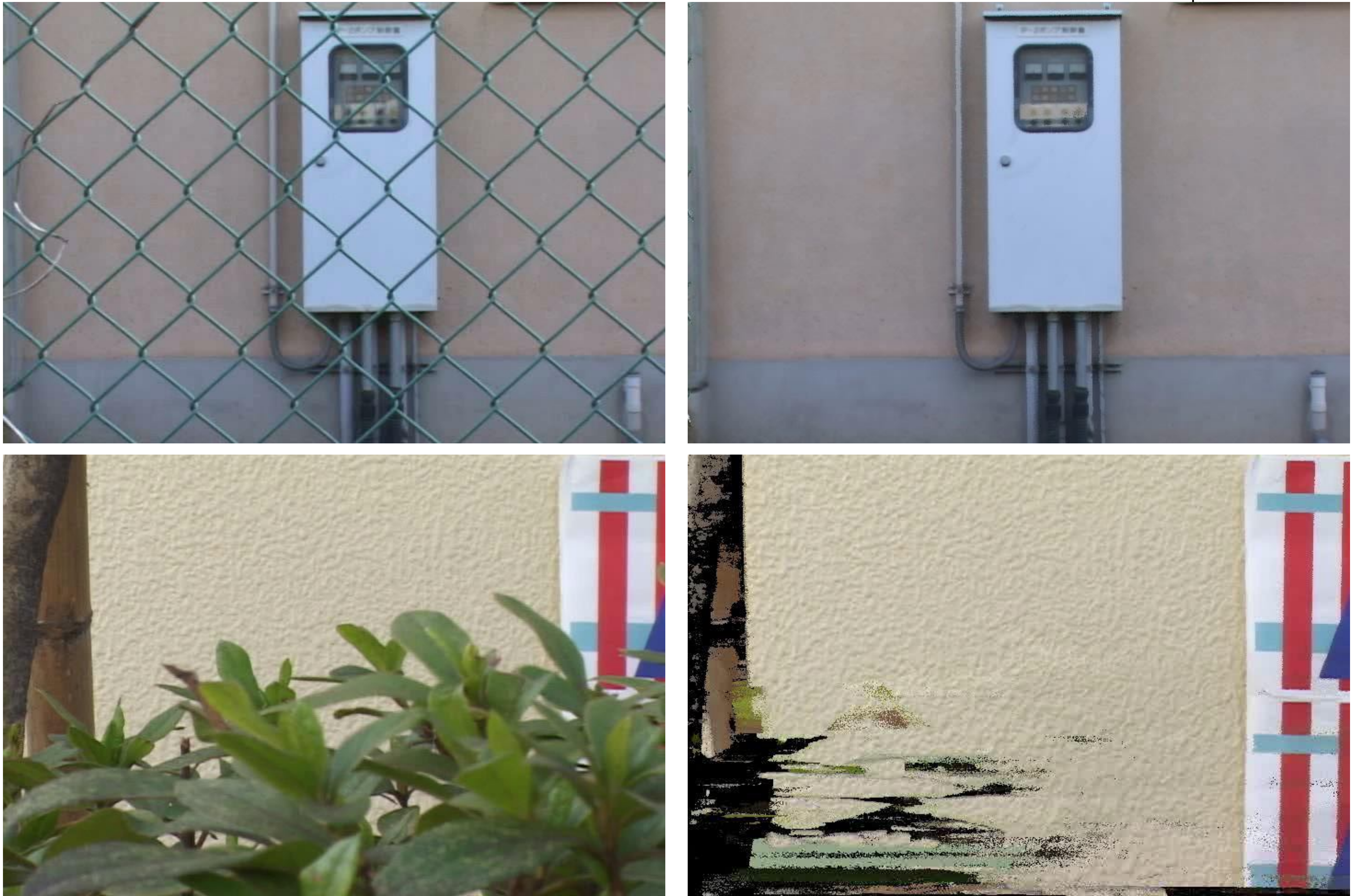
Epipolar Plane Image

- The cross section which put image and cut in epipolar line



Removal result

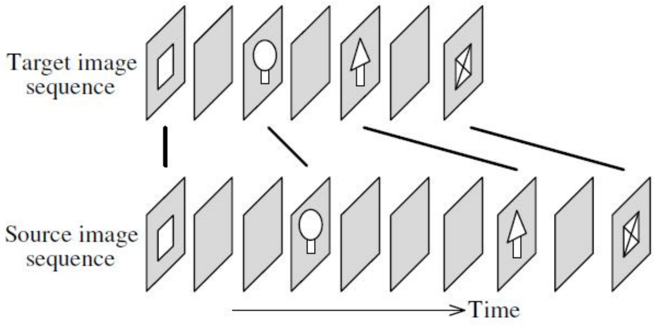




Removing Foreground Objects [Uchiyama 2010]

No need for assuming that the scene is composed of a set of planar structure

Use multiple video stream,
Stitch the background region
(Foreground = Moving object)



Frame-to-frame matching is already done by DP

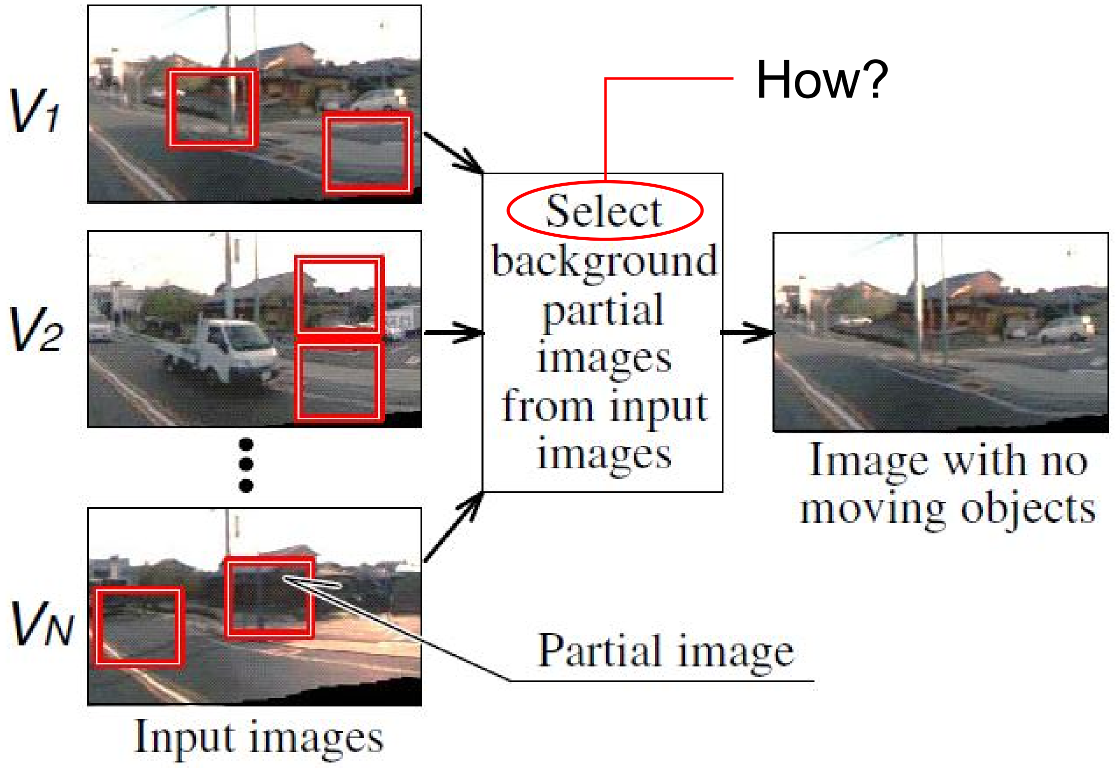
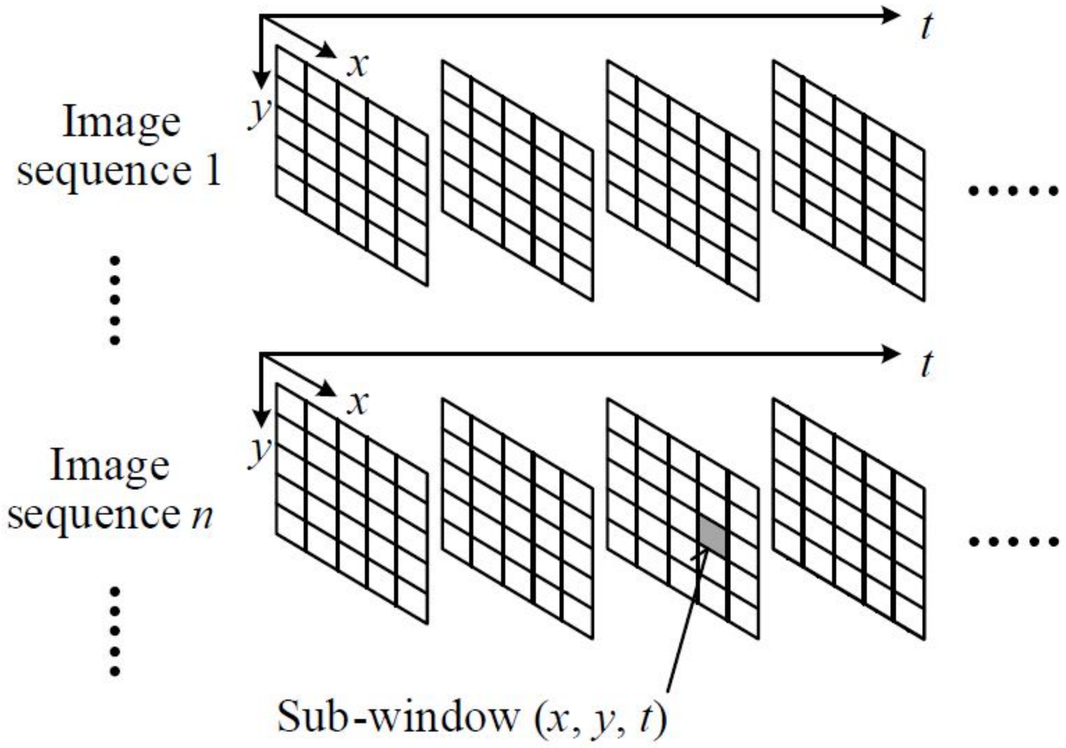

How?
V1
V2
VN
Select background partial images from input images
Image with no moving objects
Partial image
Input images

Figure 2. Omni-directional camera image containing no moving object is obtained from many images captured at the same place in a different timing independently.

Background Selection

Idea: Background is

1. Observed most often throughout all video streams
2. Consistent between neighboring sub-windows

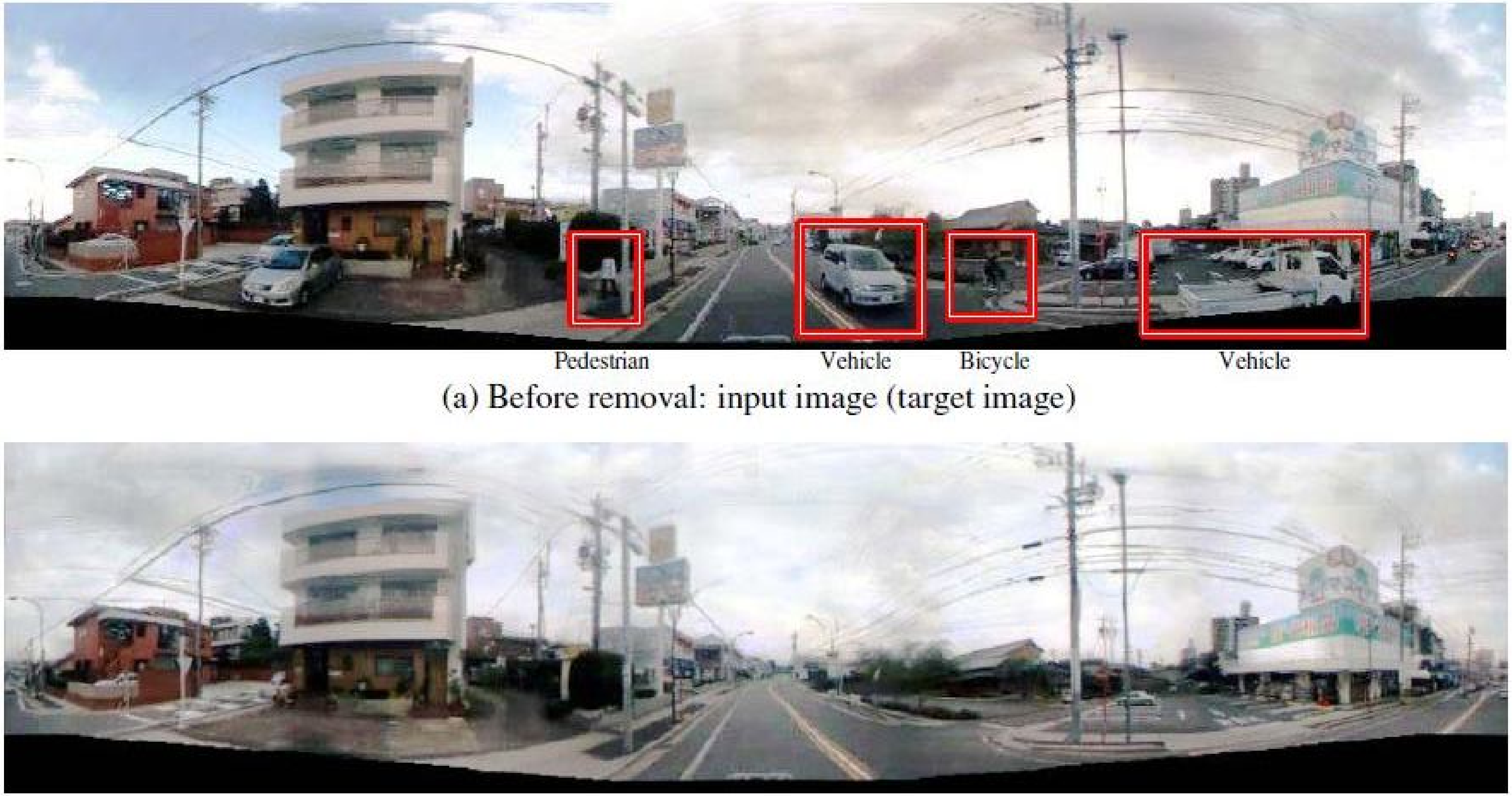


Vector median
(Color median)

$$\underset{\mathbf{v}\in\{\mathbf{v}_1,\ldots,\mathbf{v}_N\}}{\arg\min} \sum_{i=1}^{N} |\mathbf{v} - \mathbf{v}_i|$$

i: stream ID

Removing Foreground Objects [Uchiyama 2010]



(a) Before removal: input image (target image)

(b) After removal: output image

Figure 7. Result of the proposed method. Although a pedestrian, vehicles and a bicycle are observed in the input image (a), they were removed in the output image (b).

Video Summarization

東京大学
THE UNIVERSITY OF TOKYO

Video Synposis
[Rav-Acha 2006]

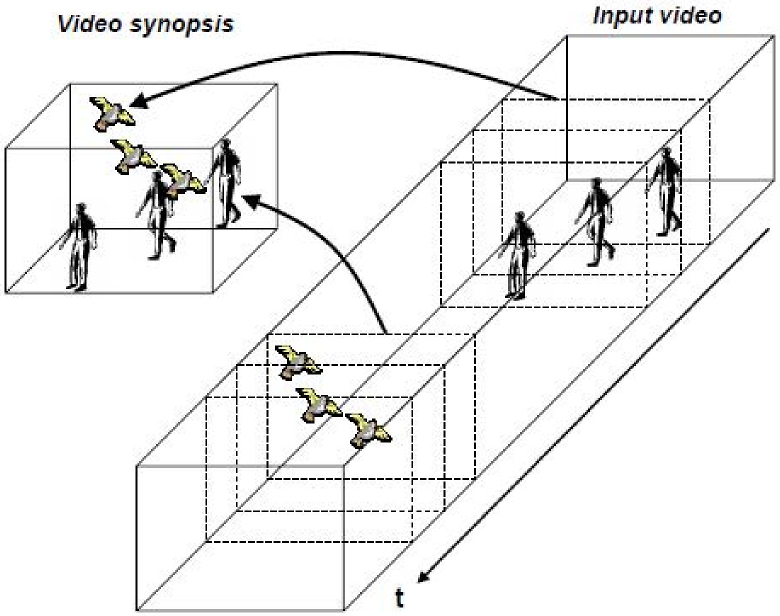


Figure 1. The input video shows a walking person, and after a period of inactivity displays a flying bird. A compact video synopsis can be produced by playing the bird and the person simultaneously.

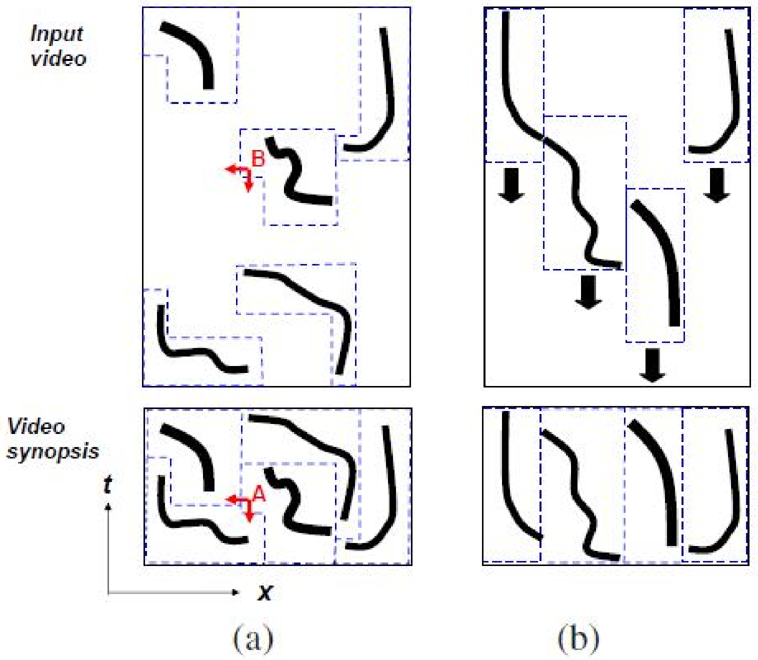
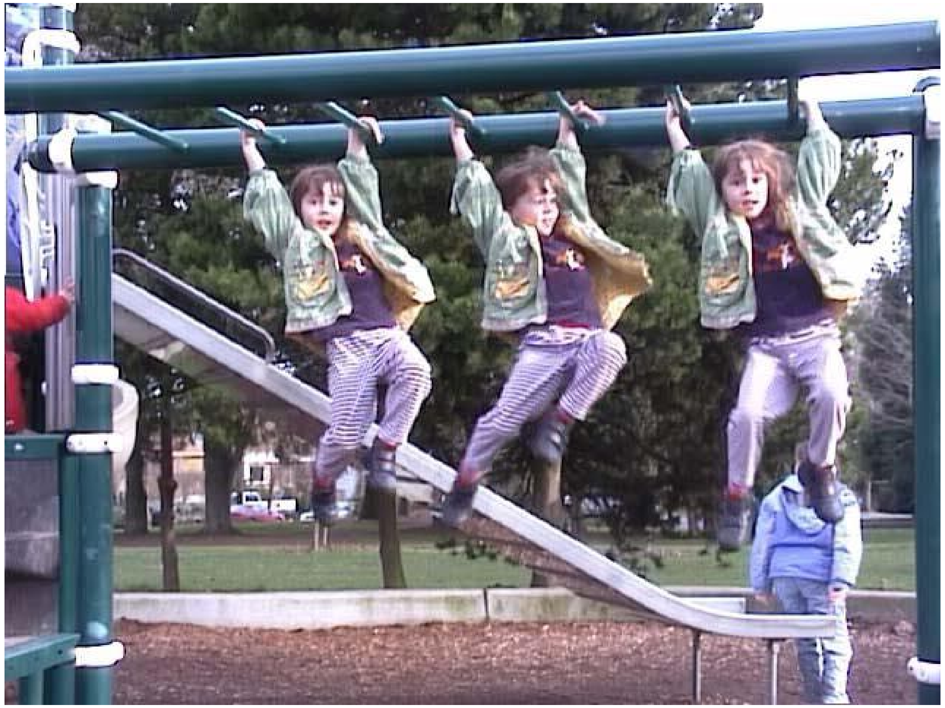


Figure 2. In this space-time representation of video, moving objects created the “activity strips”. The upper part represents the original video, while the lower part represents the video synopsis. (a) The shorter video synopsis S is generated from the input video I by including most active pixels. To assure smoothness, when pixel A in S corresponds to pixel B in I, their “cross border” neighbors should be similar.
(b) Consecutive pixels in the synopsis video are restricted to come from consecutive input pixels.

Result

http://www.vision.huji.ac.il/video-synopsis/



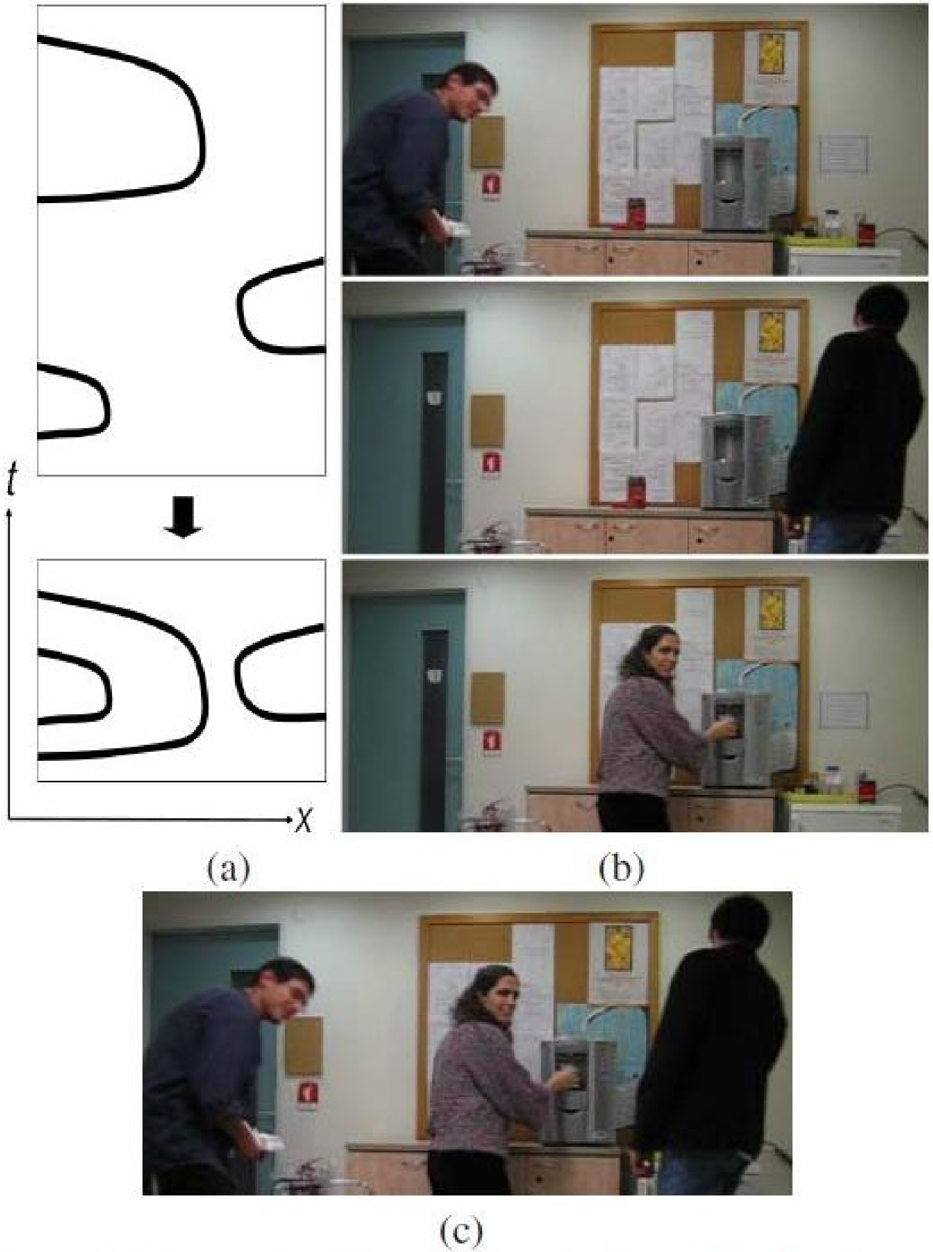
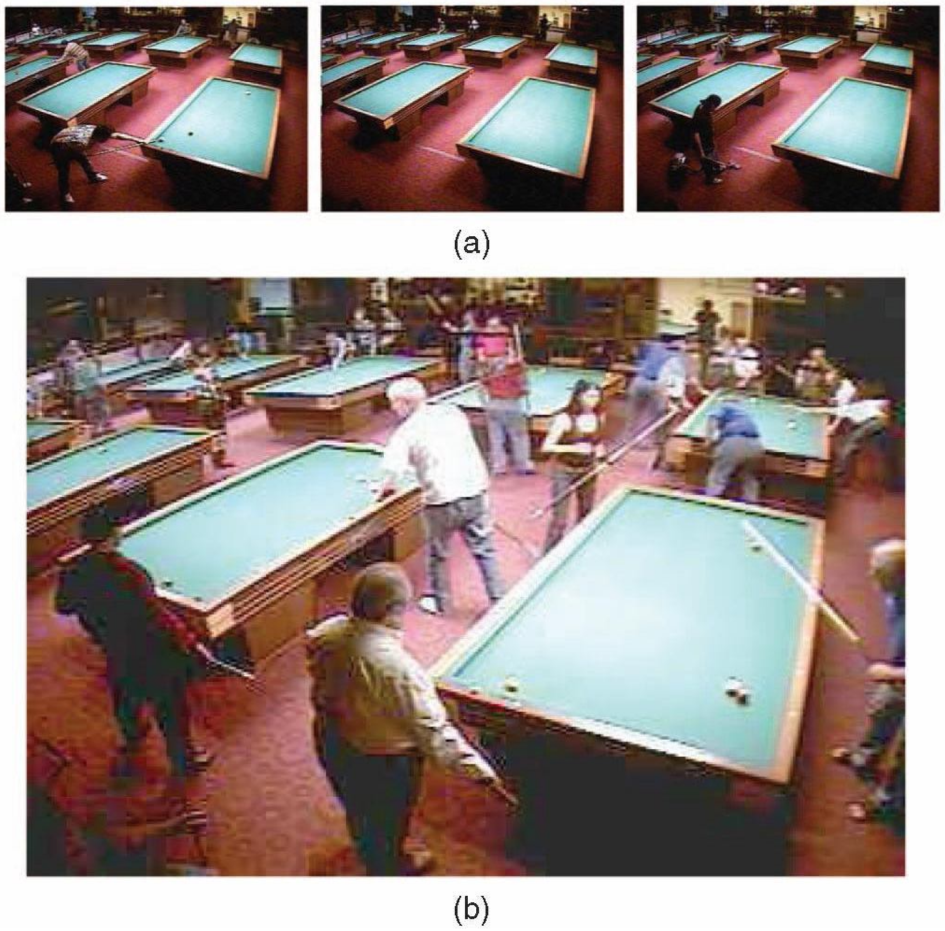
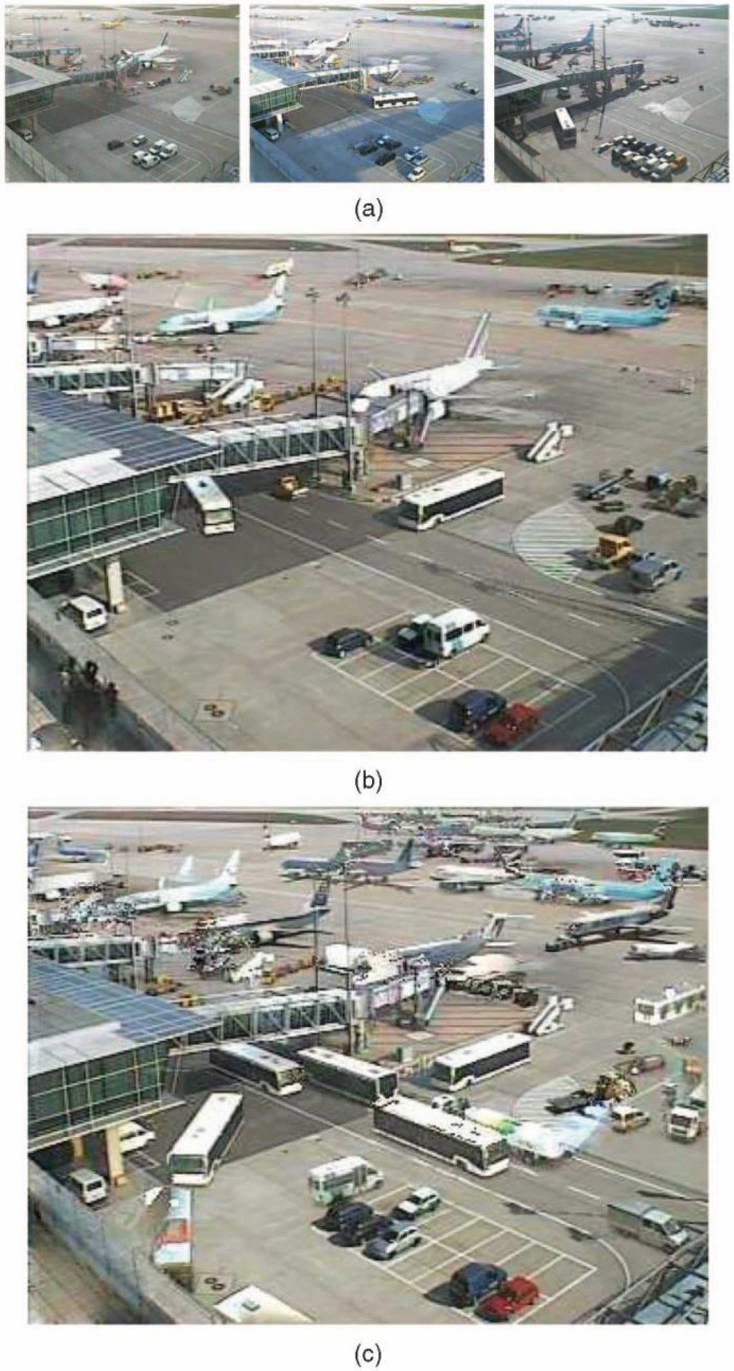


Figure 6. An example when a short synopsis can describe a longer sequence with no loss of activity and without the stroboscopic effect. Three objects can be time shifted to play simultaneously. (a) The schematic space-time diagram of the original video (top) and the video synopsis (bottom). (b) Three frames from original video. (c) One frame from the synopsis video.



(a)

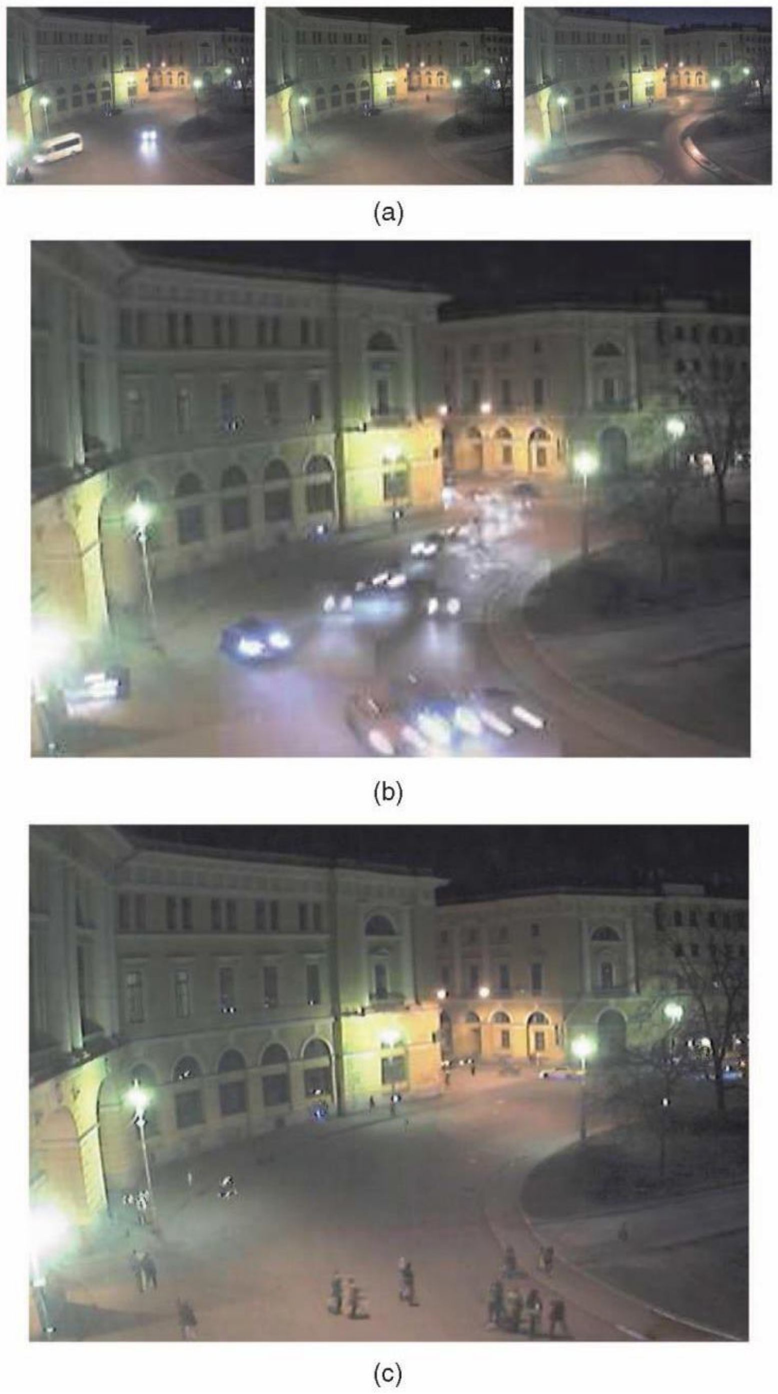
(b)



(a)

(b)

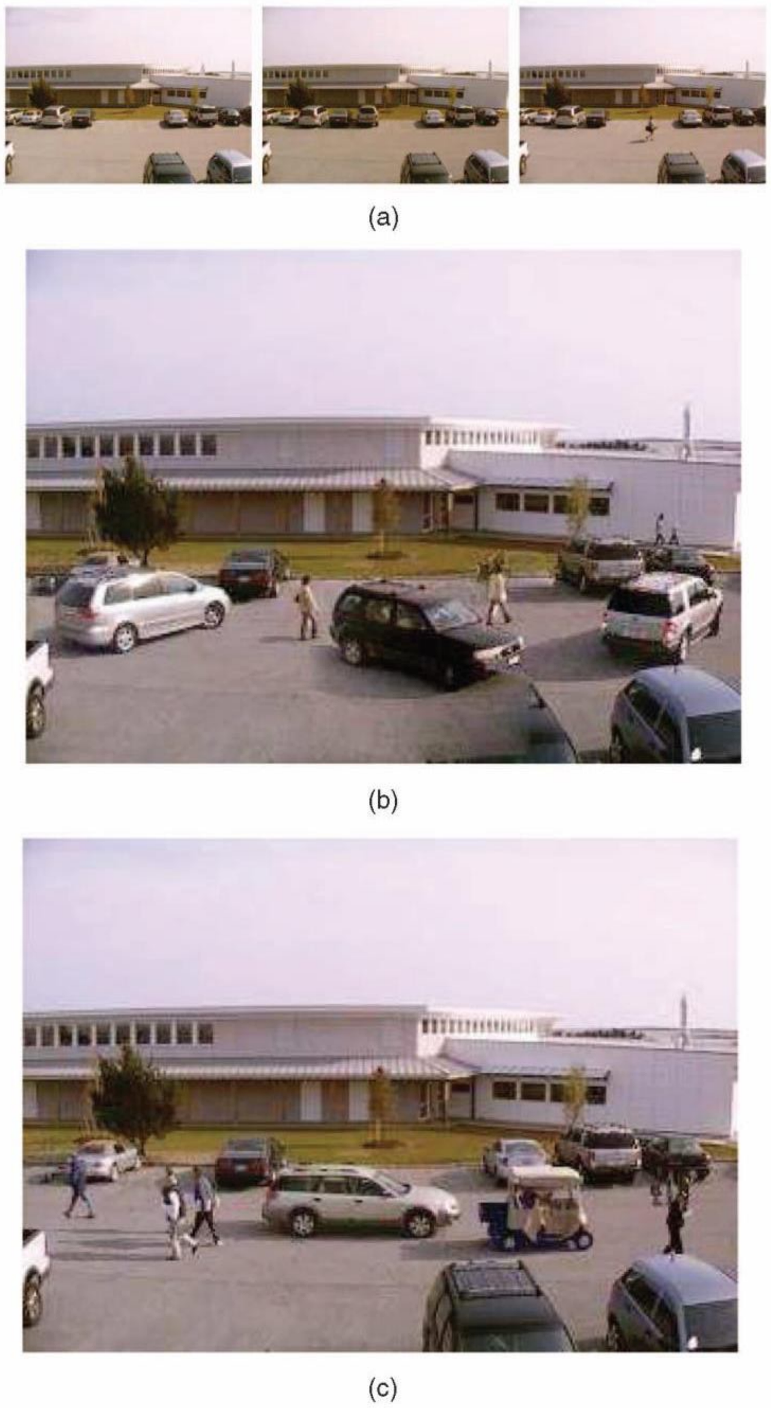
(c)



(a)

(b)

(c)



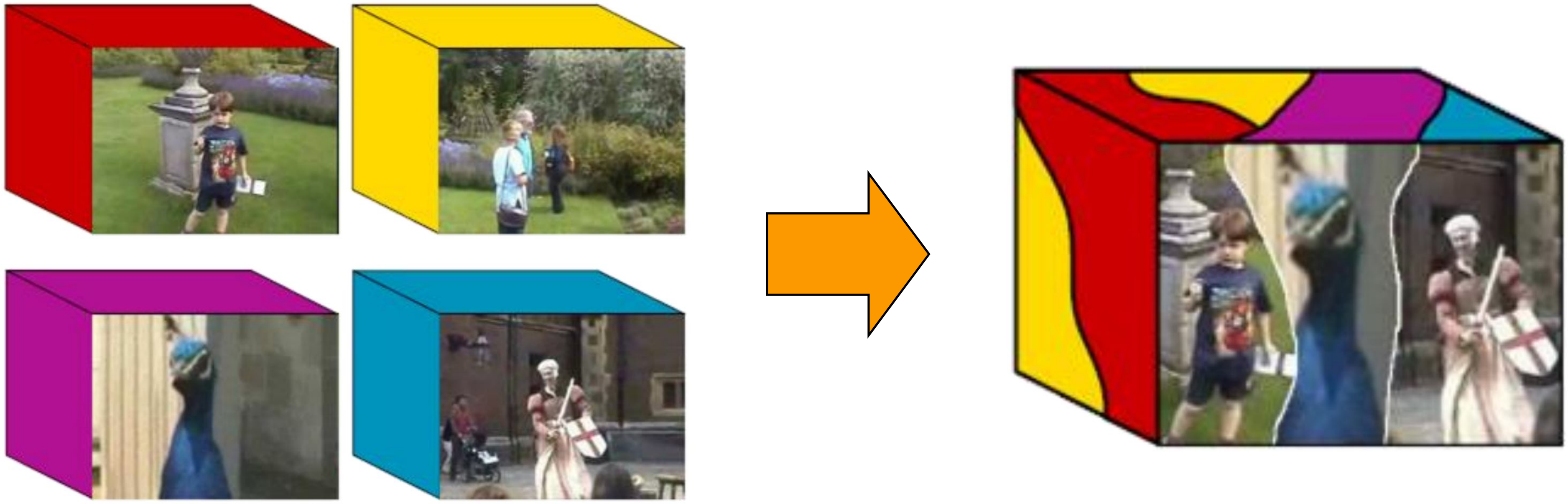
(a)

(b)

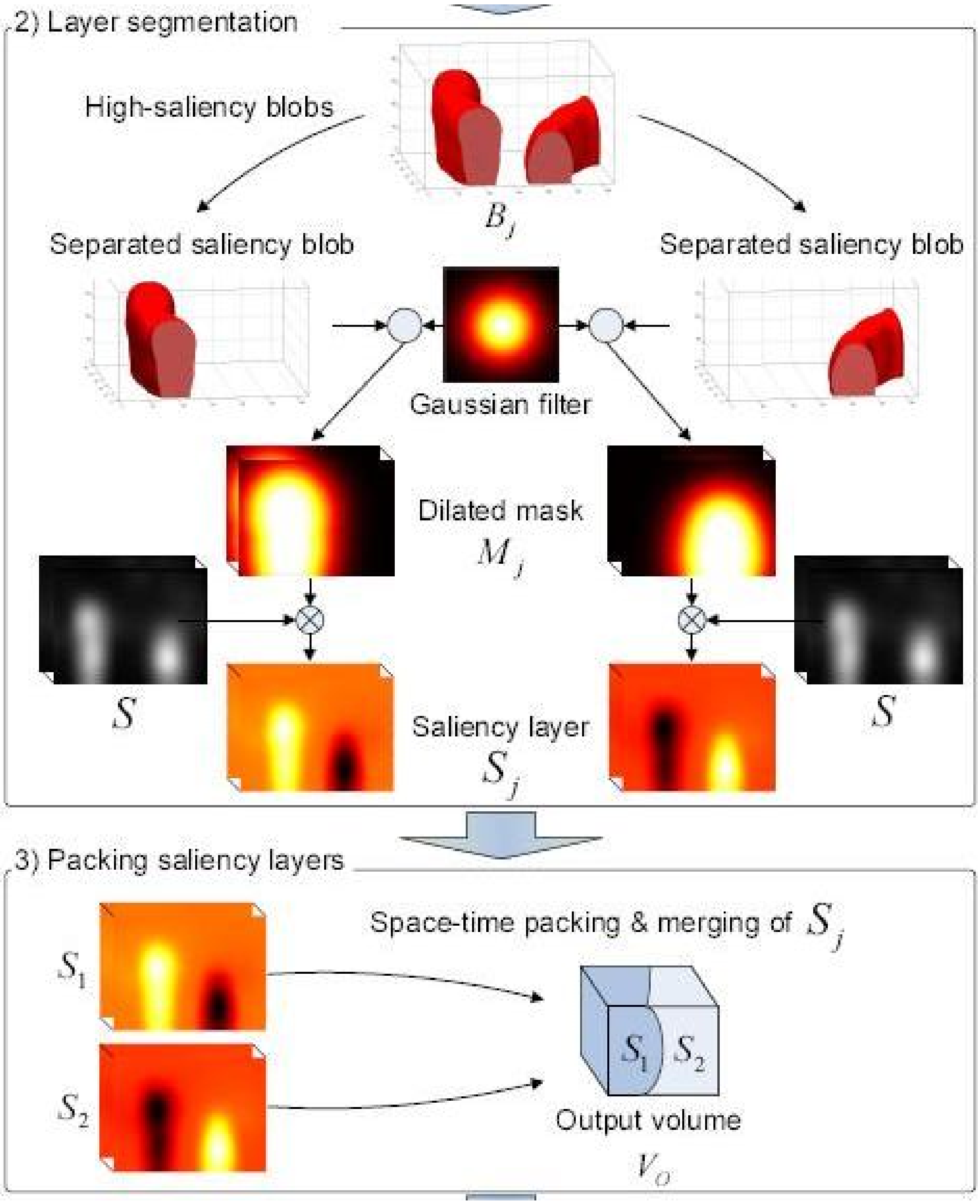
(c)

Space-time Video Montage
(H.W. Kang 2006)

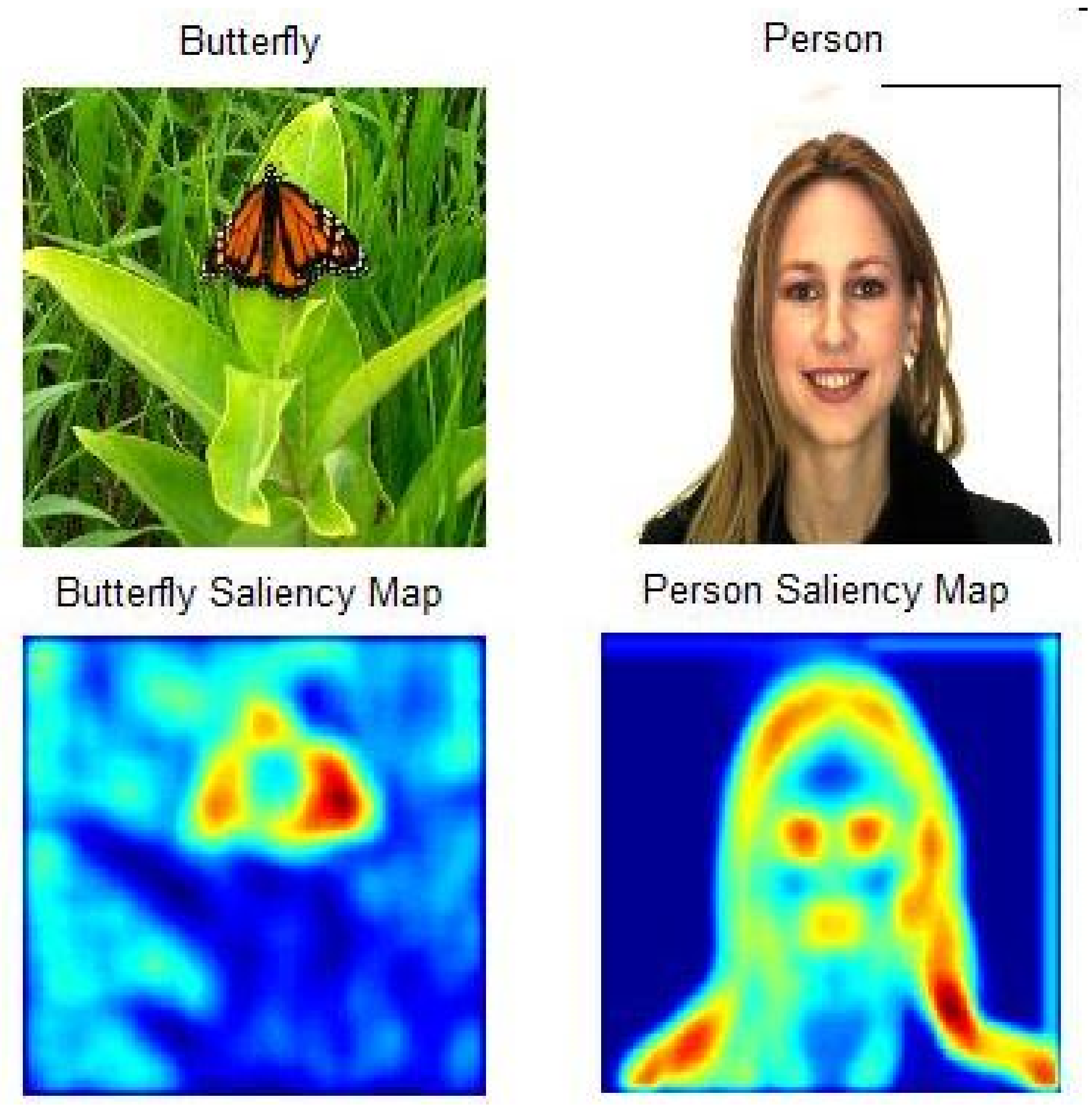
- Video summarization based on space-time analysis
- Define “important” portions inside a volume
- Leave them, exclude others



Details


2) Layer segmentation
High-saliency blobs
B_j
Separated saliency blob
Separated saliency blob
Gaussian filter
Dilated mask
M_j
S
S
Saliency layer
S_j
3) Packing saliency layers
Space-time packing & merging of S_j
S_1
S_2
S_1
S_2
Output volume
V_O

Saliency?
(顯著性)

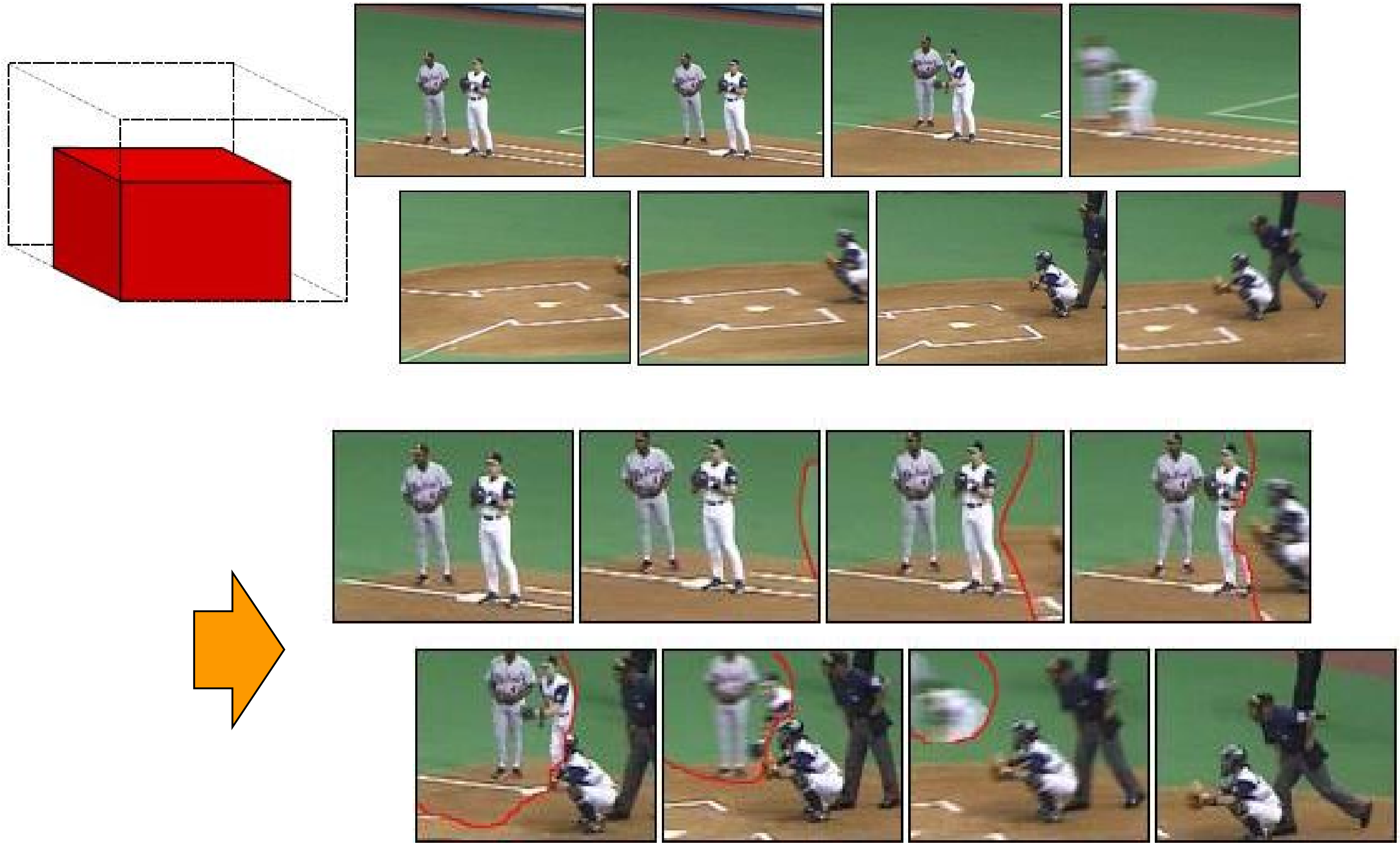


[MathWorks (MATLAB)]

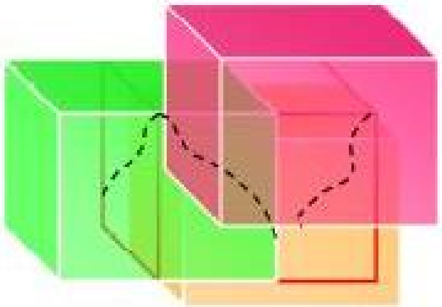
Simple example:

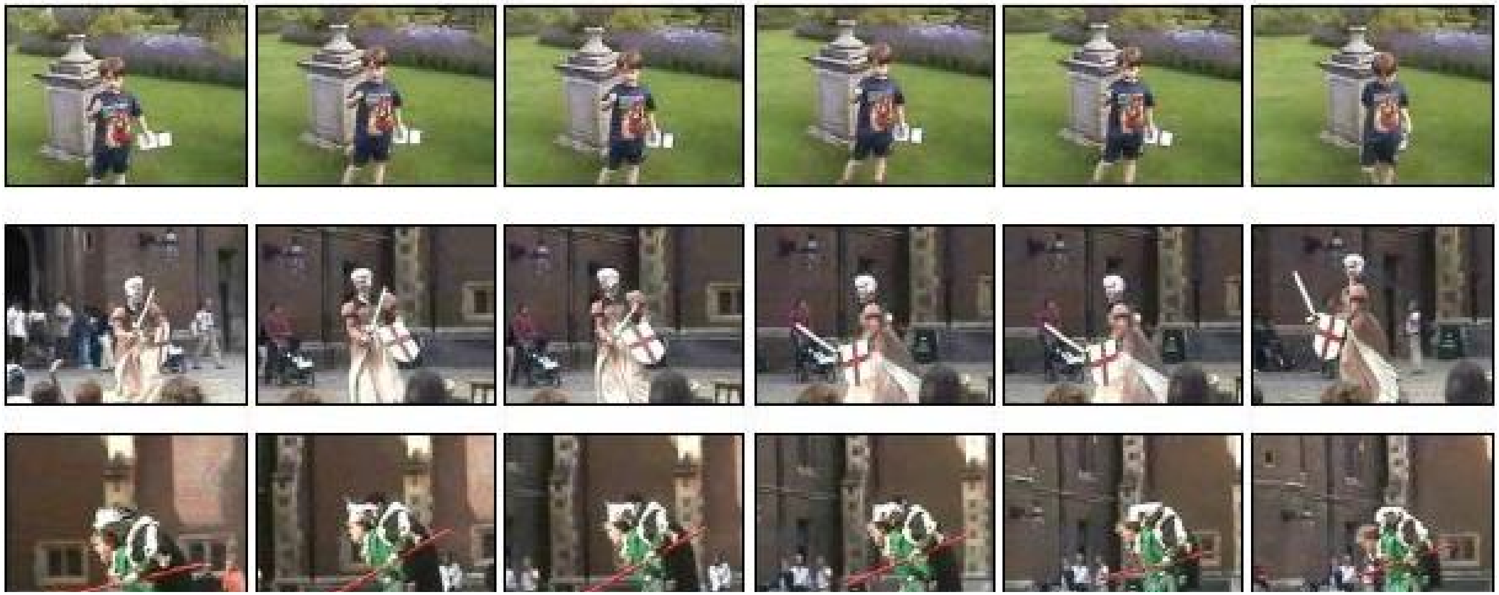
Difference between
Original image and
Gauss-filtered image

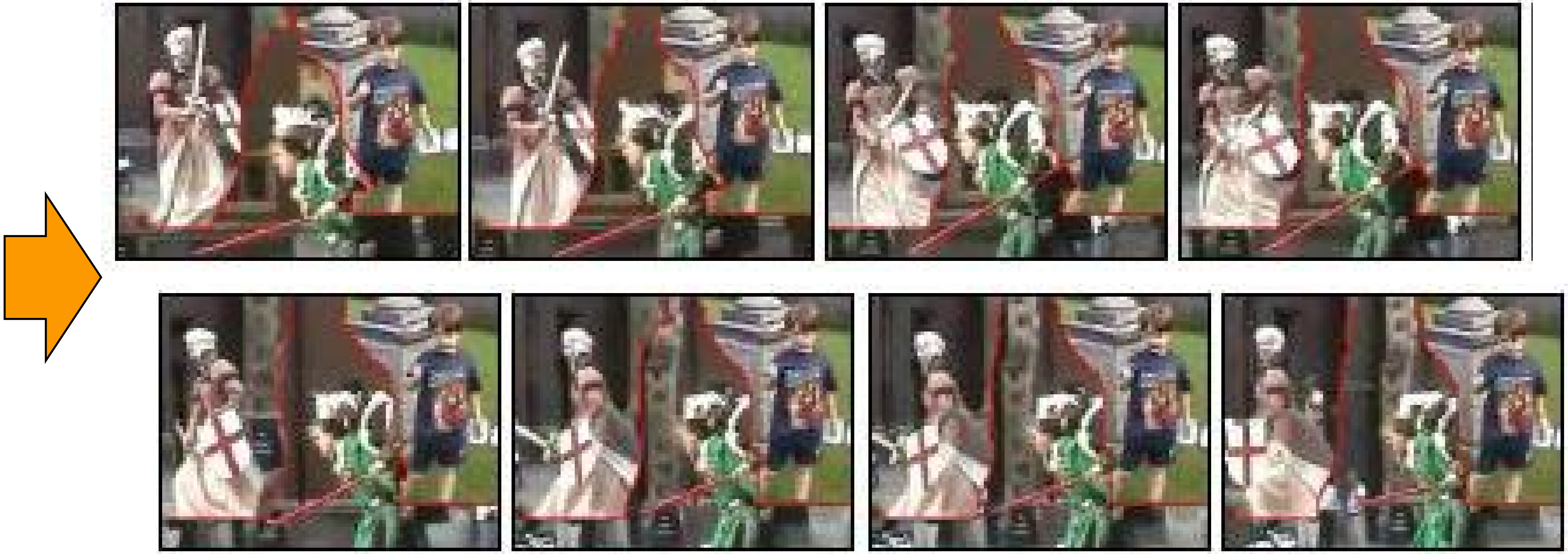
Result



Result





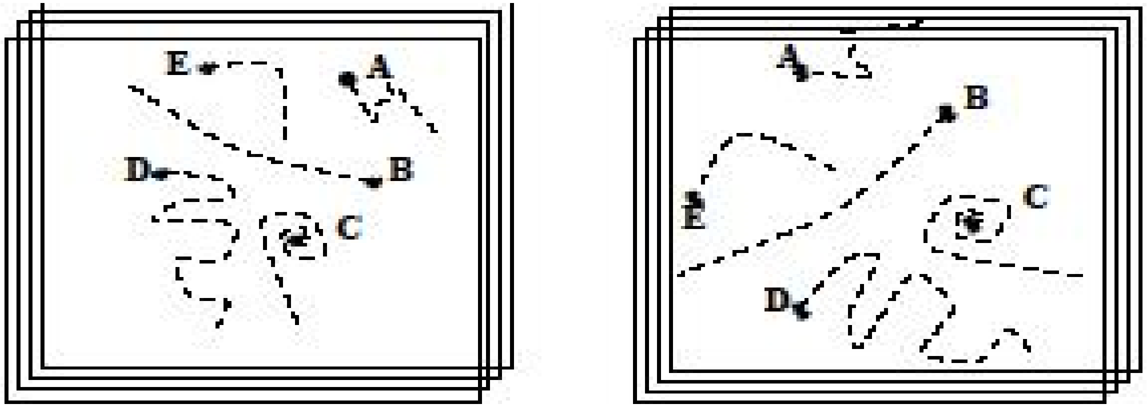


Video Joint

東京大学
THE UNIVERSITY OF TOKYO

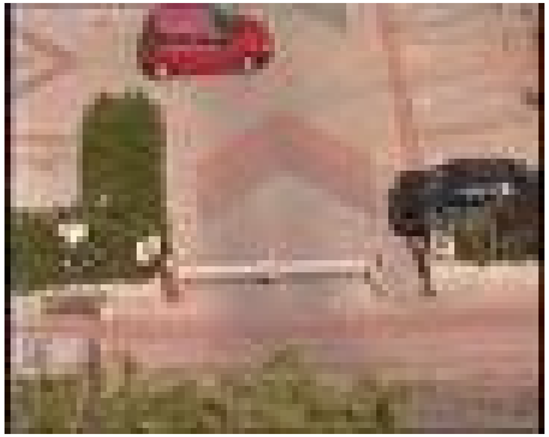
Feature-Based Video Alignment
(Irani 2006)

Problem formulation:
Cameras are static → Estimate homography H(3x3) and temporal deviation △t

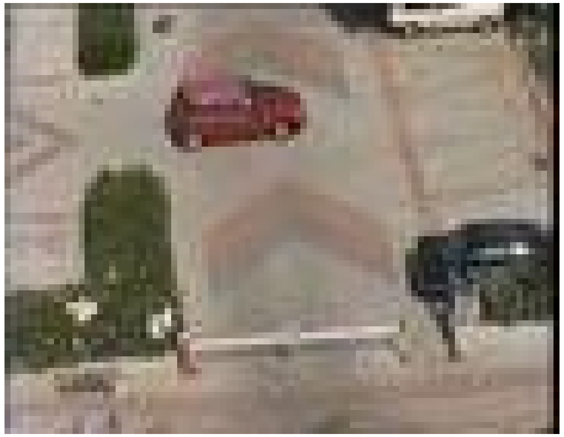


$$\vec{P}(H,\Delta t) = \arg\min \sum_{trajectories} \| (x_1, y_1, t) - H(x_2, y_2, t+\Delta t) \|^2$$

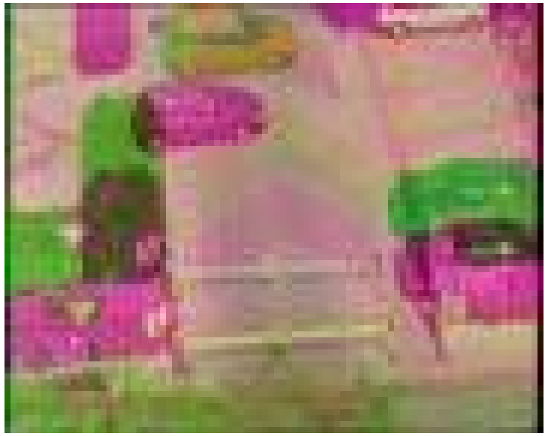
Feature-Based Video Alignment:
An example



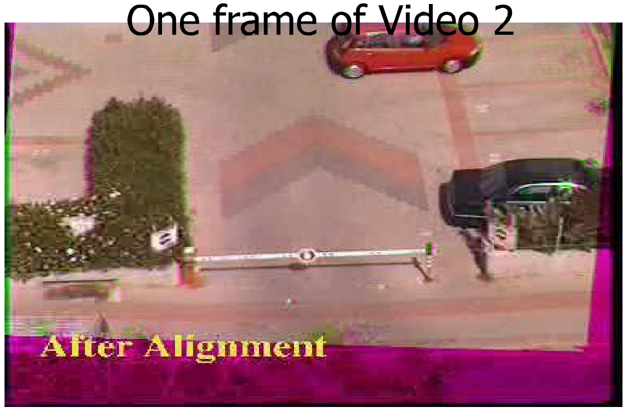
One frame of Video 1



One frame of Video 2



Before alignment



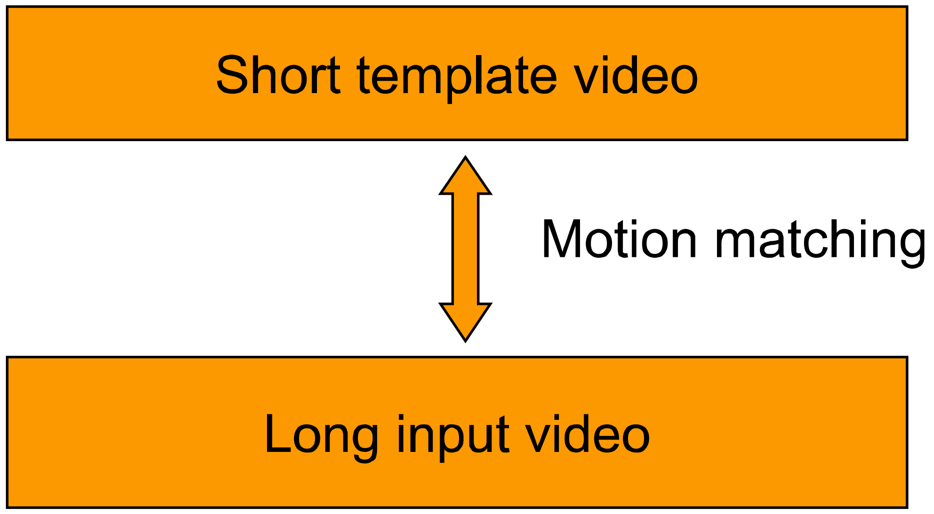
After alignment

Behavior Analysis

東京大学
THE UNIVERSITY OF TOKYO

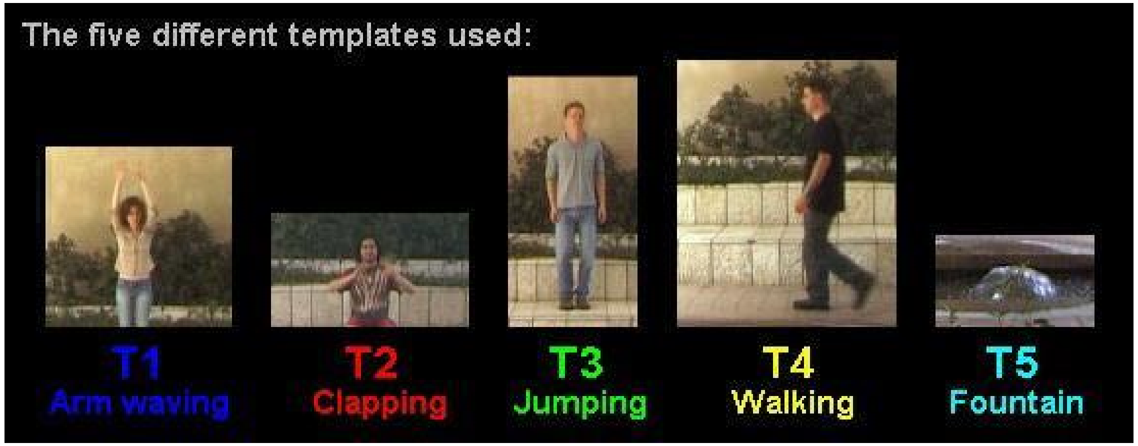
Space-Time Behavior Based Correlation (E. Shechtman* 2005, 2007)

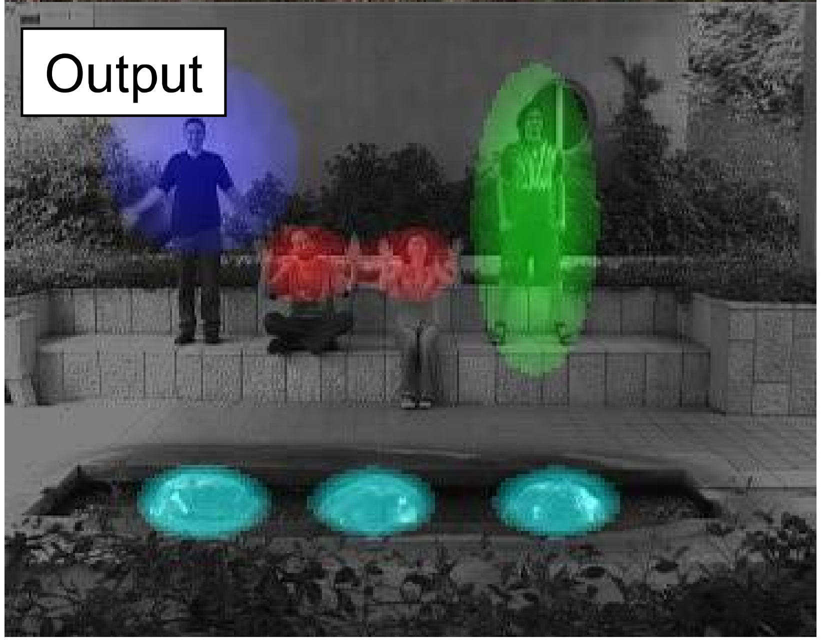
- Extract similar behavior
- By calculating correlation between portion & portion inside a S-T volume



Space-Time Behavior Based Correlation
[Irani et al. 2005 (CVPR)]

Template Video

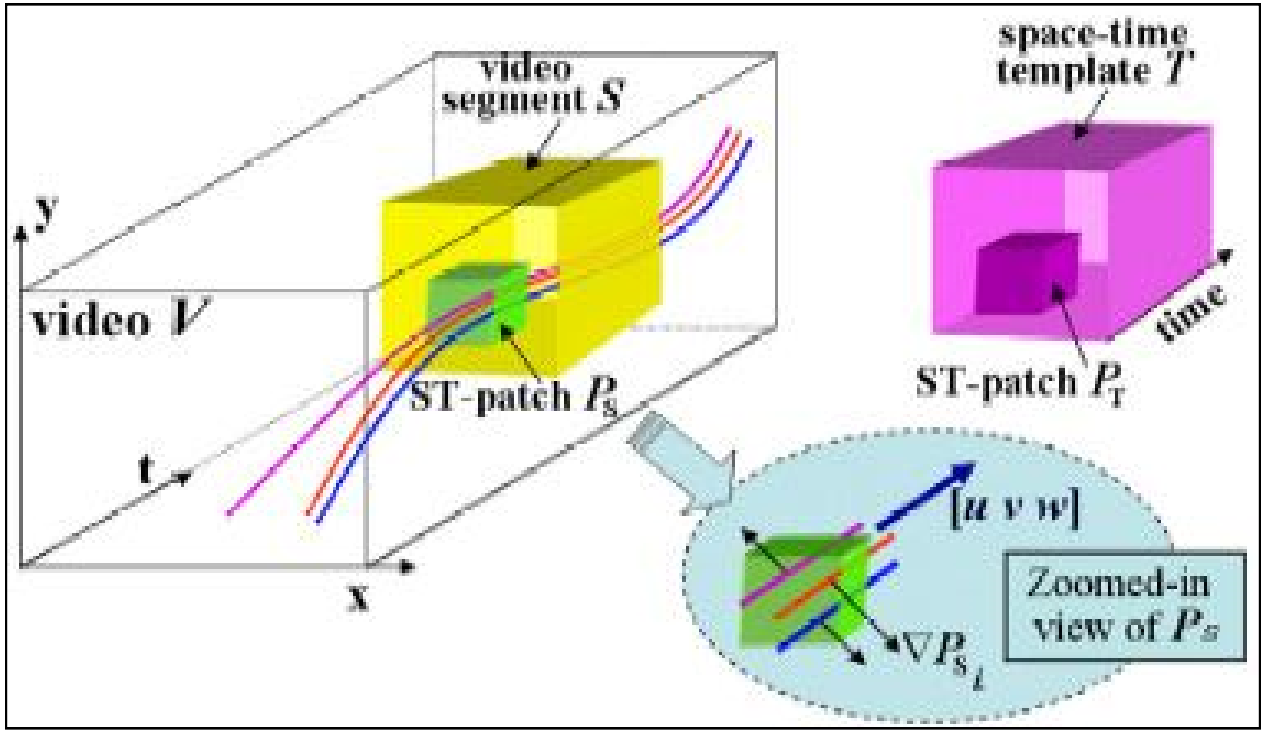




Features:
- 3D-block matching in Space-Time Volume
- Recognize different behaviors simultaneously!

Space-Time Behavior Based Correlation

Property of Space-Time patch



T Template video

S Same size as *T* in input video

Small patch

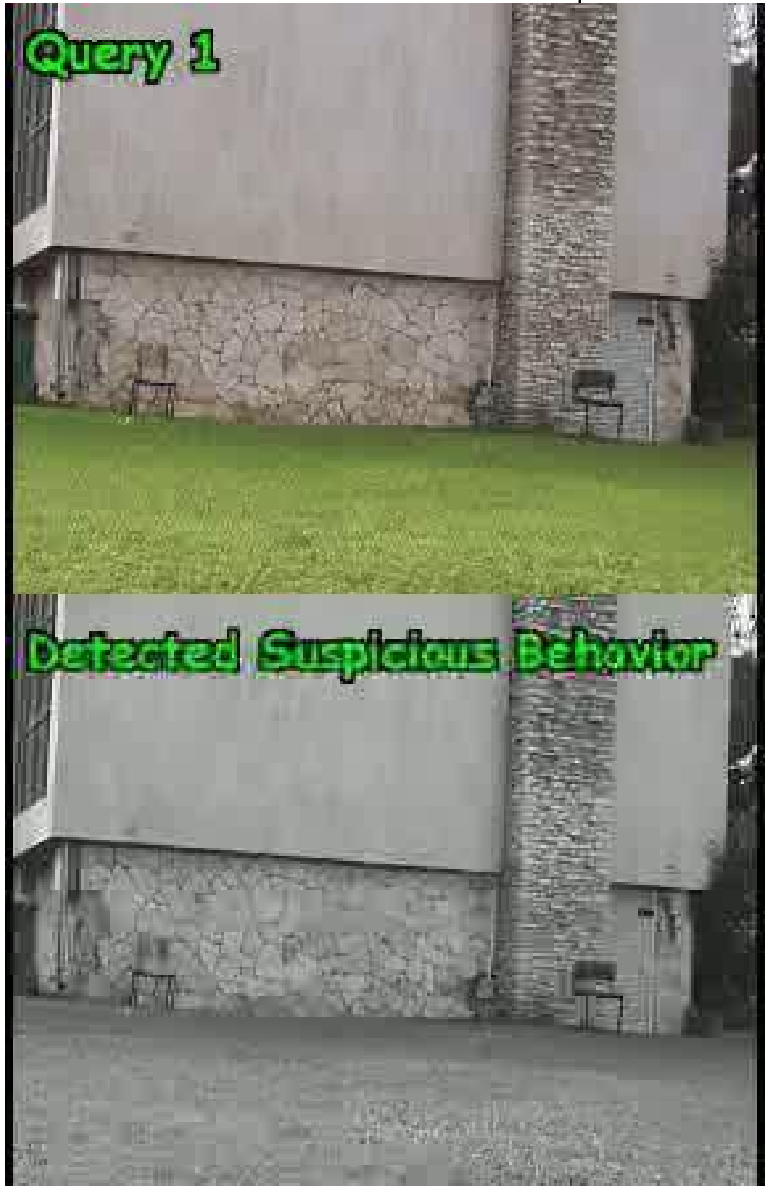
Space-time gradients of color value *P*

$$\nabla P_i = (P_{xi}, P_{yi}, P_t) \perp (u, v, w) \Rightarrow \nabla P \begin{bmatrix} u \\ v \\ w \end{bmatrix} = 0 \Rightarrow \begin{bmatrix} p_{x1} & p_{y1} & p_{t1} \\ & \cdots & \\ p_{xn} & p_{yn} & p_{tn} \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}_{nx1}$$

Detecting Irregularities in Images and in Video (O. Boiman* 2005)

A portion which is not similar to any portion inside the database volume

Irregular



Others

東京大学
THE UNIVERSITY OF TOKYO

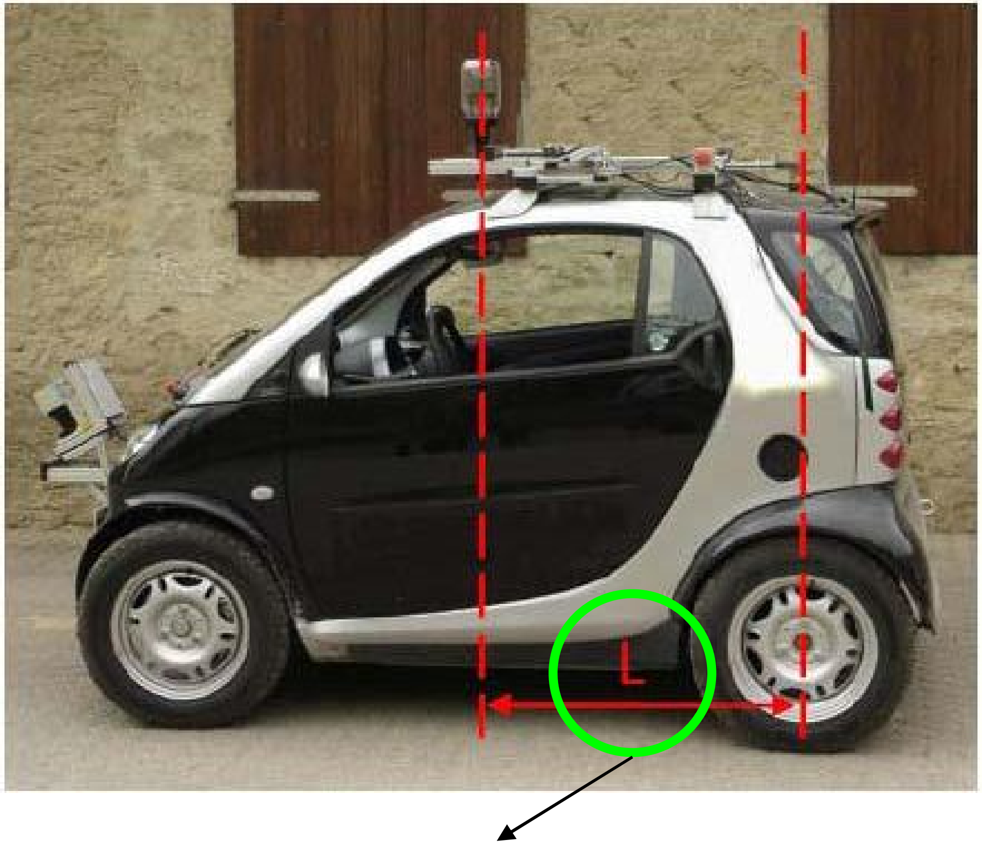
Absolute Scale in SfM from a Single Vehicle Mounted Camera

[Scaramuzza 2009]

- **SFM (Visual Odometry) from a single camera:**
  - The absolute scale is unknown
- **When the camera is mounted on a wheeled vehicle:**
  - The absolute scale can be recovered
  - Very accurately, and fully automatically
- **Because**
  - Wheeled vehicles undergo local circular motion

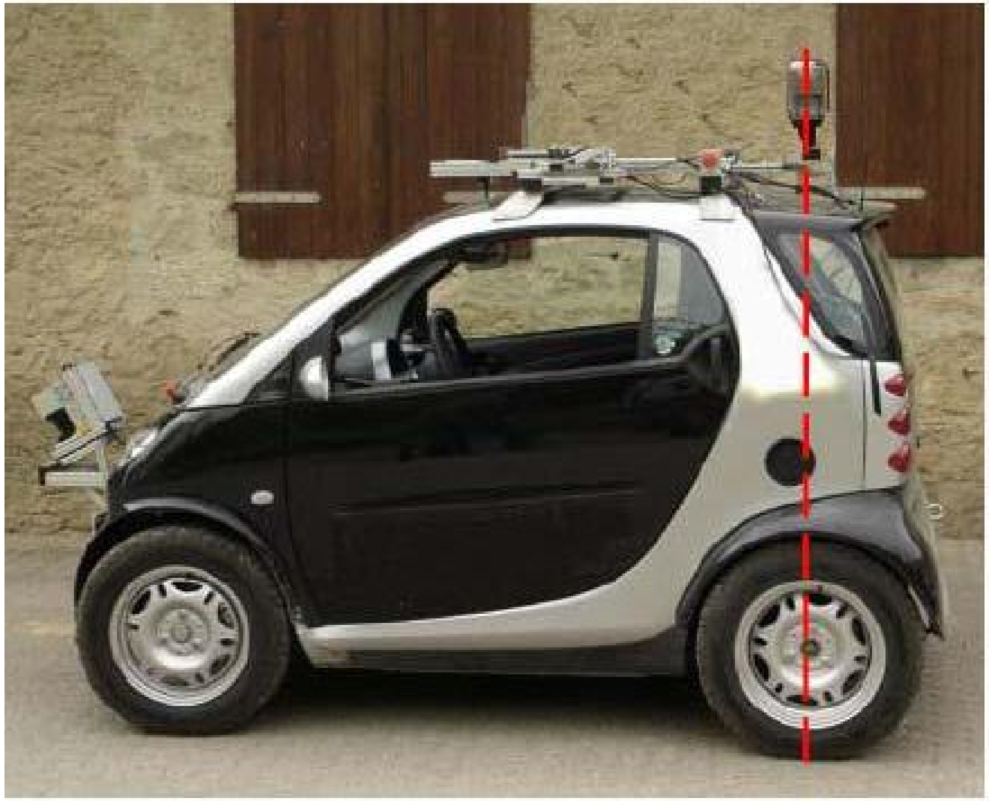
Idea

Offset from non-steering axle



Gives absolute scale [ICCV09]

No offset from non-steering axle



Simplify motion estimation [ICRA09]

Motion Model of nonholonomic vehicles

- **“Nonholonomic”**

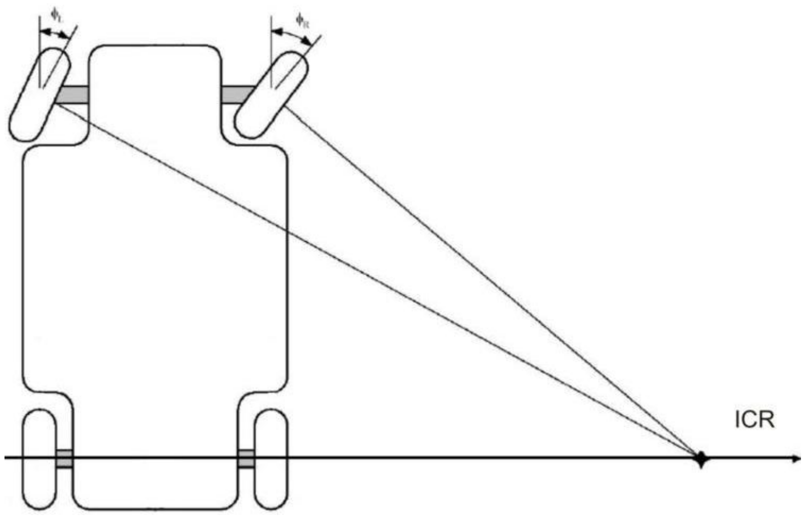
  Controllable DOF < Effective DOF

- **Example:**
  - Cars, bikes, wheel chairs, …
  - Most mobile robots, …
  - Controllable: 2 DOF

    Acceleration (1) + Steering (1)
  - Effective: 3 DOF

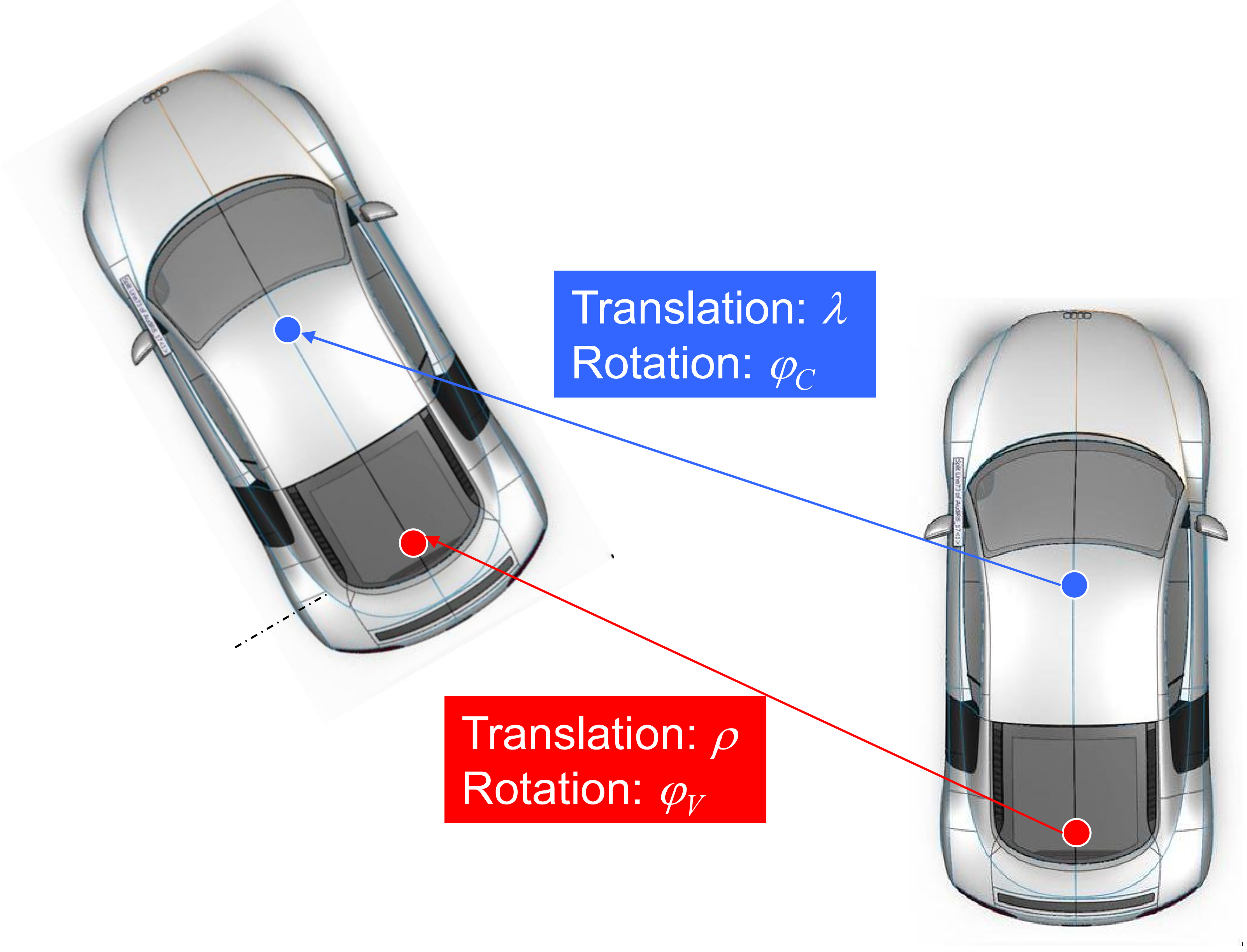
    Position (2) + Orientation (1)

- **Ackermann Steering Principle**



Wheel of the vehicle follows a circular course

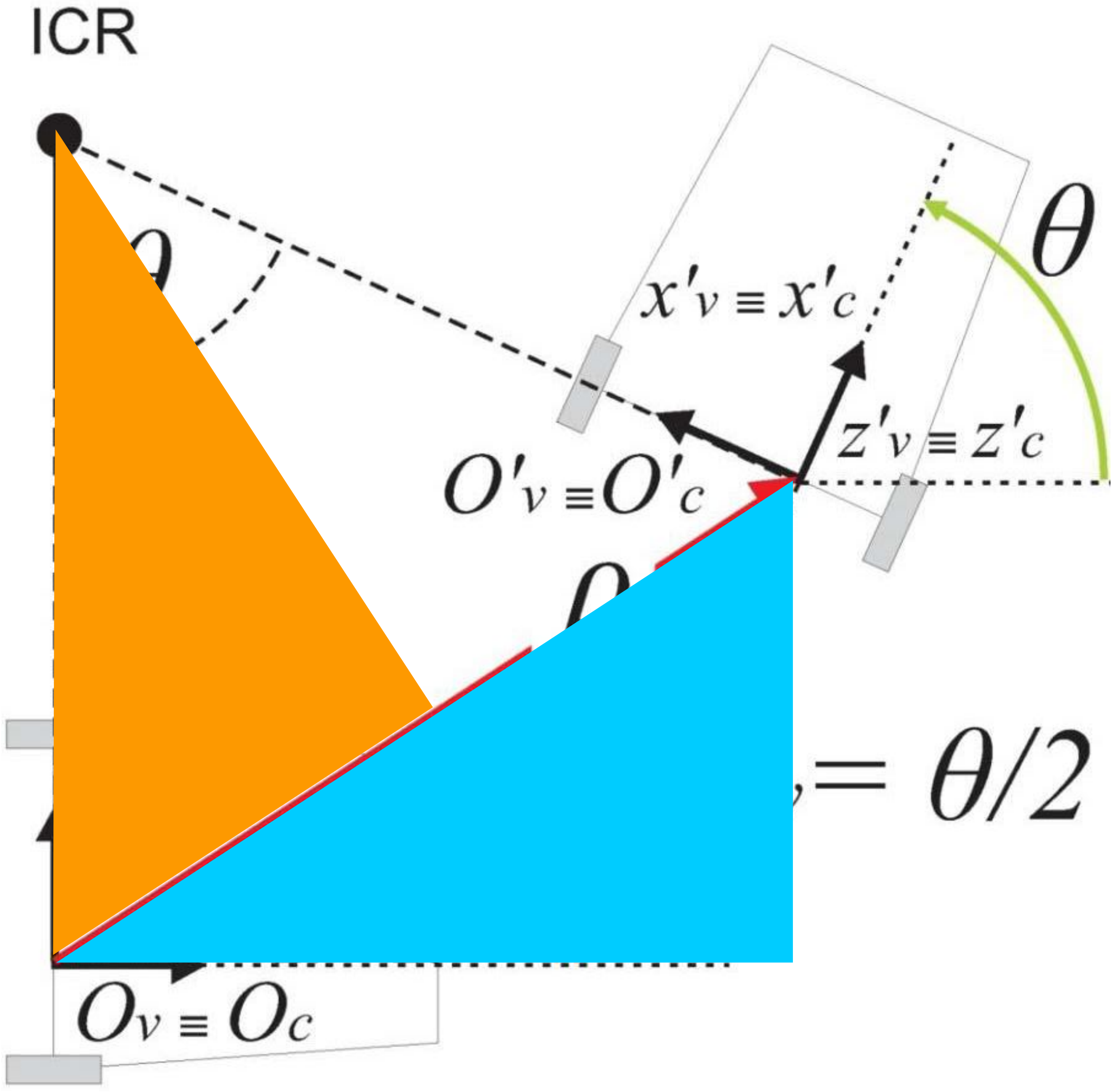
Variable Definition



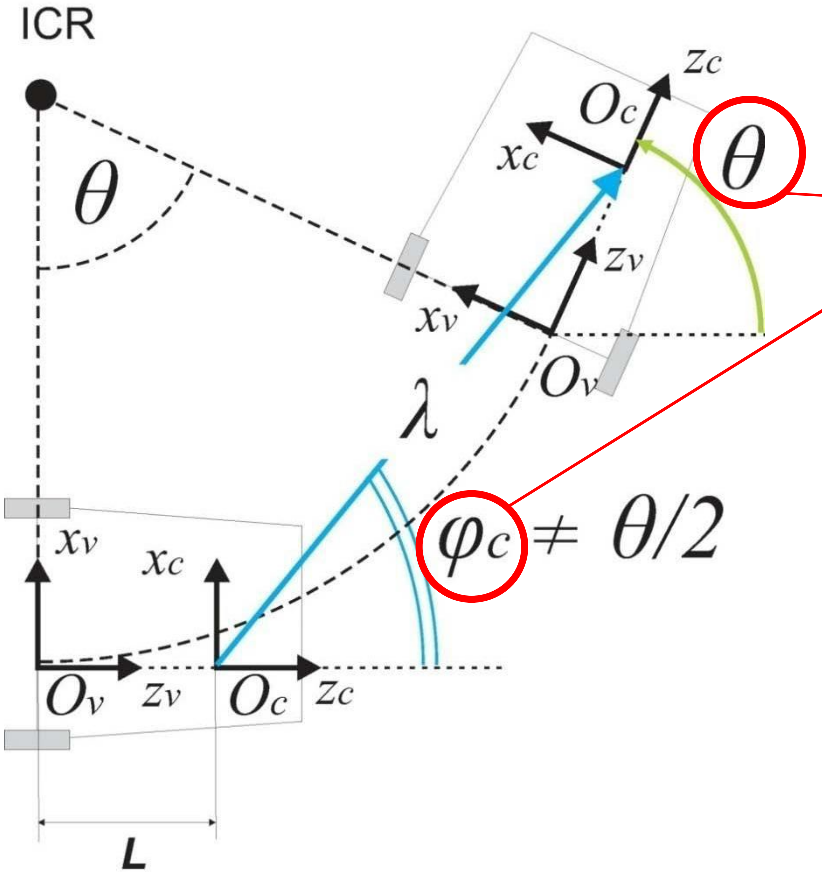
Translation: λ
Rotation: φ_C

Translation: ρ
Rotation: φ_V

Circular Motion (no offset)



Circular Motion (with offset)



Parameters can be estimated from image feature correspondences

Core equations

$$\lambda = \frac{2L\sin(\frac{\theta}{2})}{\sin(\varphi_c - \frac{\theta}{2})}$$

$$\rho = \frac{L\sin(\varphi_c) - L\sin(\varphi_c - \theta)}{\sin(\varphi_c - \frac{\theta}{2})}$$

$$E = \lambda \begin{bmatrix} 0 & \cos(\theta - \varphi_c) & 0 \\ -\cos(\varphi_c) & 0 & \sin(\varphi_c) \\ 0 & \sin(\theta - \varphi_c) & 0 \end{bmatrix}$$

Motion Estimation

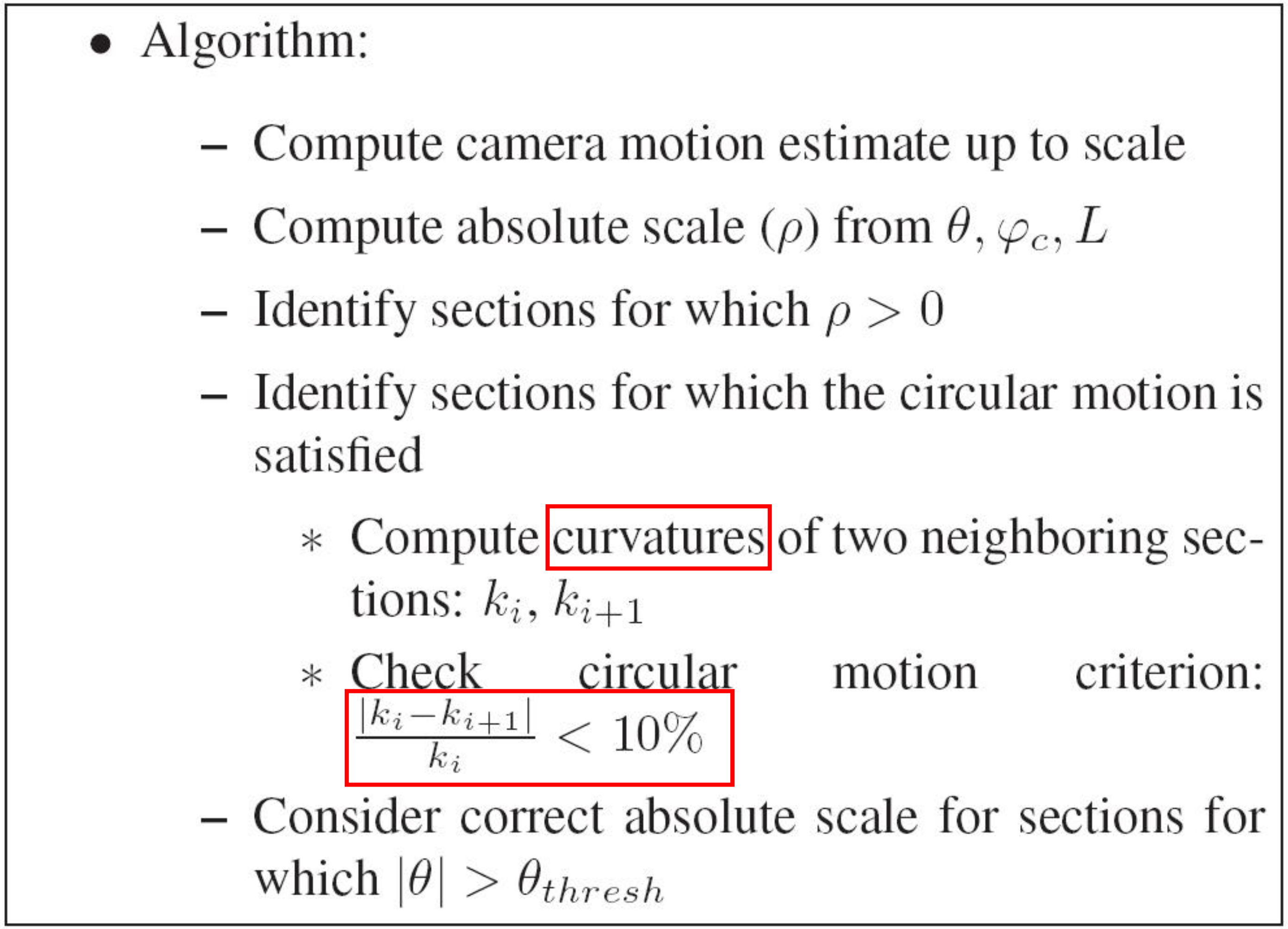
Known (image feature correspondences)

$$\begin{cases} p'^T E p = 0 \\ E = \lambda \begin{bmatrix} 0 & \cos(\theta - \varphi_c) & 0 \\ -\cos(\varphi_c) & 0 & \sin(\varphi_c) \\ 0 & \sin(\theta - \varphi_c) & 0 \end{bmatrix} \end{cases}$$

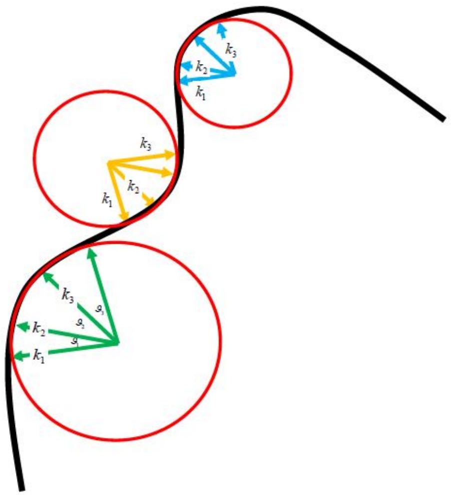
- **Method 1: Least-squares**
  - $f(\cos(), \cos(), \sin(), \sin()) = 0$
  - At least 3 point correspondences to find a solution

- **Method 2: Nonlinear**
  - Taylor expansion $g(\theta, \varphi) = 0$ & Newton’s iterative method
  - At least 2 point correspondences to find a solution

Finding Sections of Circular Motion in a Camera Path



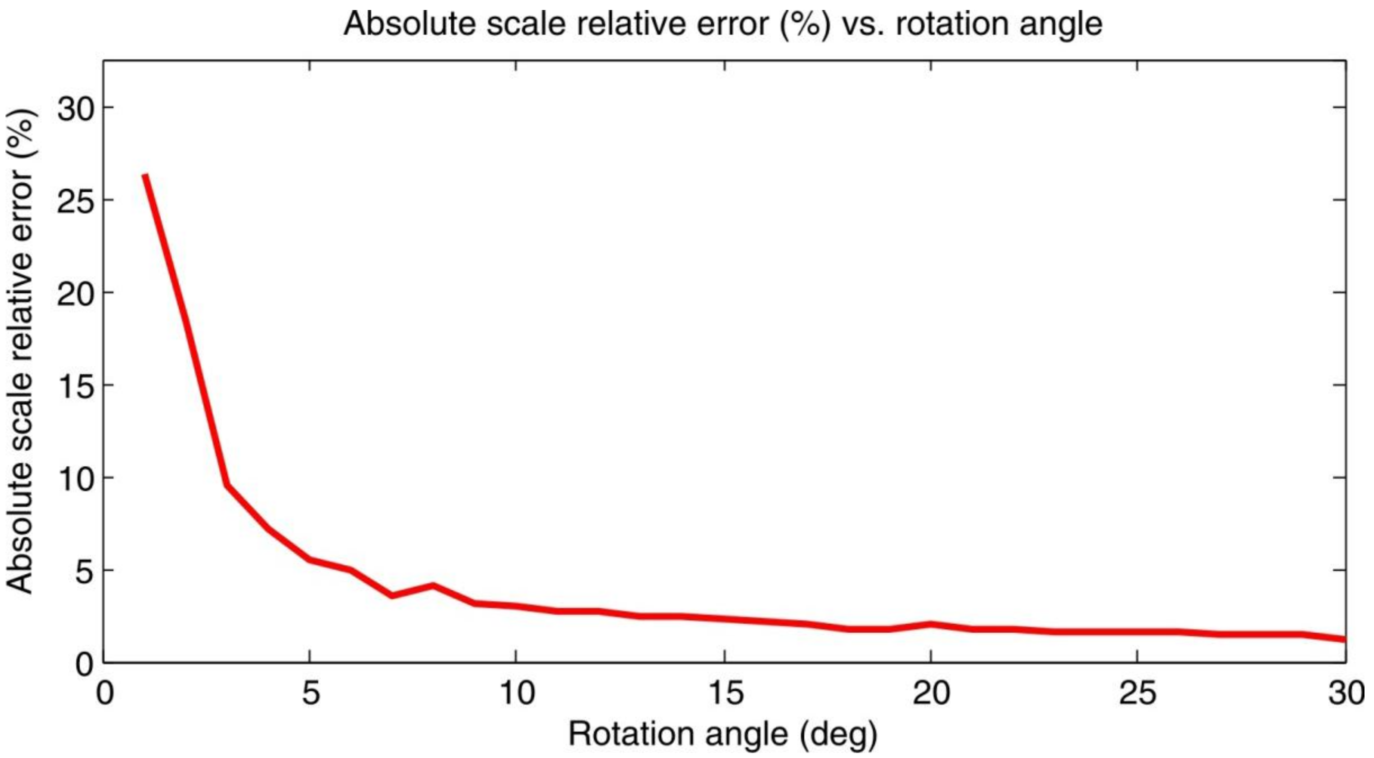
- Algorithm:
  - Compute camera motion estimate up to scale
  - Compute absolute scale (ρ) from θ, φ_c, L
  - Identify sections for which $\rho > 0$
  - Identify sections for which the circular motion is satisfied
    - ∗ Compute curvatures of two neighboring sections: k_i, k_{i+1}
    - ∗ Check circular motion criterion: $\frac{|k_i - k_{i+1}|}{k_i} < 10\%$
  - Consider correct absolute scale for sections for which $|\theta| > \theta_{thresh}$



Also
Check curvature values:
0.03 m^{-1} ~ 0.5 m^{-1}

(2 m ~ 33 m in radius)

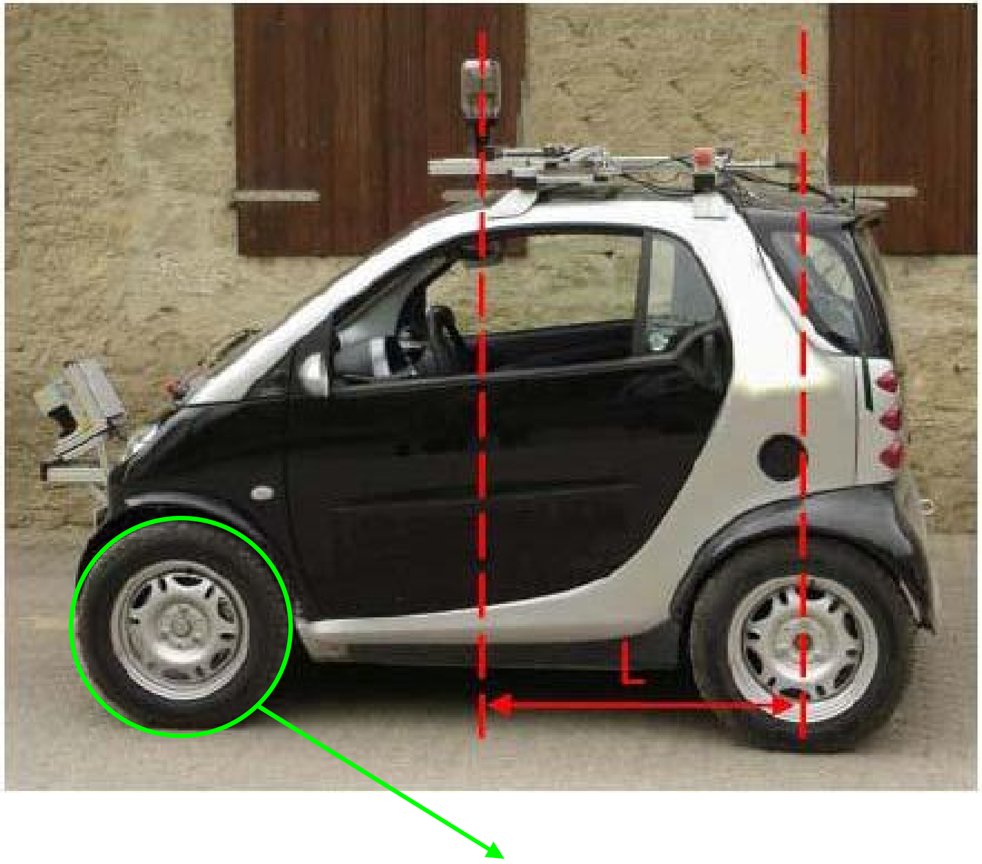
Simulation Data (+Gaussian noise): Relative Error of Absolute Scale



Absolute scale relative error (%) vs. rotation angle

- The accuracy of the scale estimate increases with θ
- The error becomes smaller than 5% for θ>10°

Real Experiment

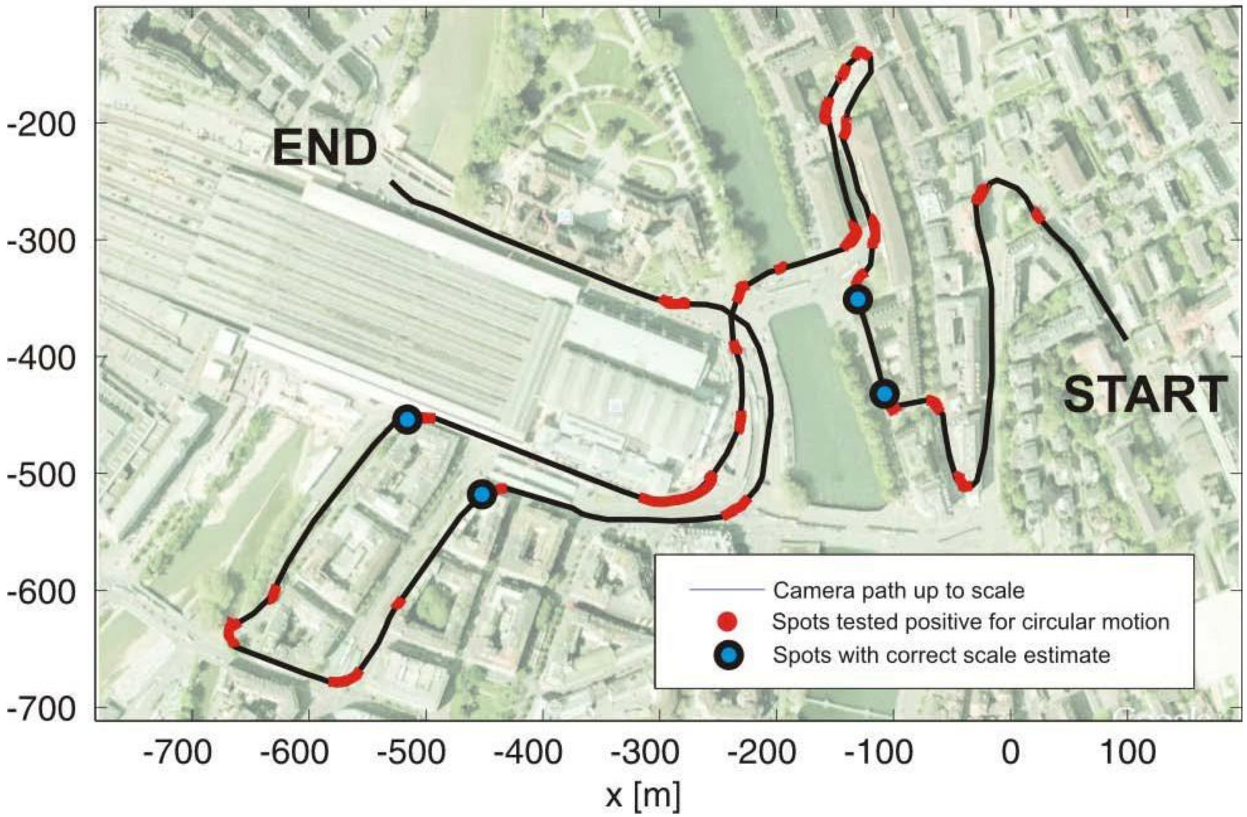


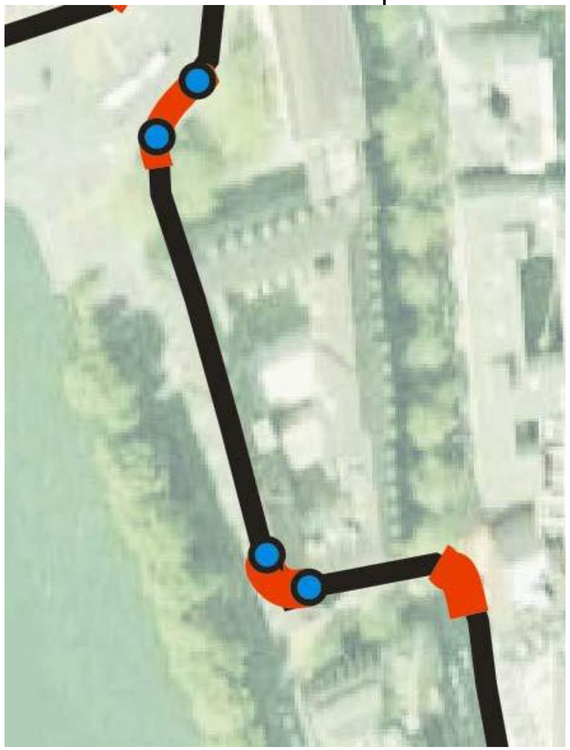
Ground truth: wheel odometry

- Offset: 0.9 m
- Omnidirectional camera (curved mirror)
- 640x480, 10 fps
- 10 ~ 45 km/h
- 3 km travel

Real Data

Comparison between visual odometry and ground truth





$\theta_{thresh} [^\circ]$	# detected	# correct
5	461	193
10	153	65
20	36	21
30	8	8

→ Within 30% of wheel odometry measurements

References

- http://research.microsoft.com/users/yasumat
- http://research.microsoft.com/~ywexler
- http://people.csail.mit.edu/celiu/motionmag
- http://www.gvu.gatech.edu/perception/projects/videotexture
- http://www.wisdom.weizmann.ac.il/~irani
- http://people.csail.mit.edu/sand/vid-match