360° Video Technology Exploration

Roger Zimmermann

FXPAL & Media Management Research Lab

NUS School of Computing

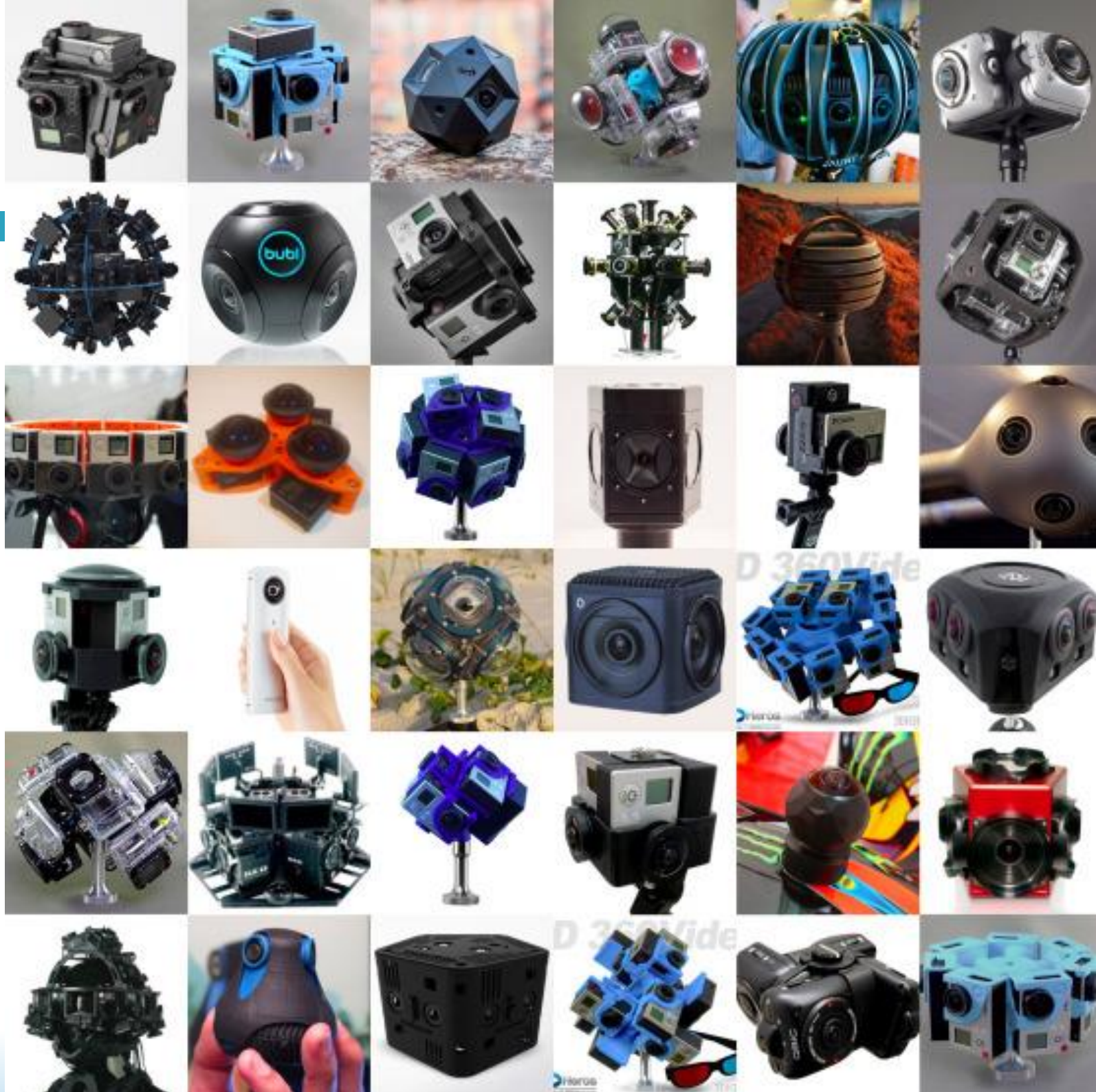
Outline

- 360° Video Hardware
- Layout Projection
- Optimizing the Streaming Delivery of 360° Videos
- Viewing Direction Determination
- Video Stabilization
- Salient Object Identification

Acknowledgements

- Lynn Wilcox, Andreas Girgensohn, Don Kimber, Chelhwon Kim, Jim Vaughan, FXPAL
- Yifang Yin, *post-doc at SoC, NUS*
- Bayan Ta'ani, *PhD student at SoC, NUS*

History



360° Video Hardware (1)

- Trend towards affordable, consumer-oriented cameras

360° Video Hardware (2)

- Samsung Gear 360 SM-C200 (2016)
  - Captures high-resolution images (up to 30MP) and video in 360-degrees
  - Dual 15MP CMOS Sensors
  - Dual f/2.0 Fisheye Lenses
  - 3840 x 1920 Video Recording at 30 fps
  - 30MP Still Images in Dual Lens Mode
  - Dual & Single Lens Modes
  - Built In Wi Fi, Bluetooth & NFC
  - Supports microSD cards up to 128GB
  - US$ 300

360° Video Hardware (3)

- Samsung Gear 360 (2017)
  - Compatible with Galaxys and iPhones
  - Supports microSD cards up to 256GB
  - Can live stream
  - US$ 175

360° Video Hardware (4)

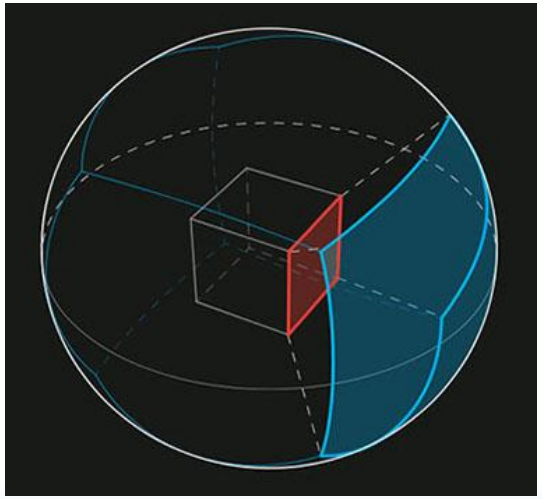
- Live Streaming with 360° cameras
- Examples (2017):
  - Giroptic iO
  - Samsung Gear 360 (2017)
  - Insta360 Nano/Air

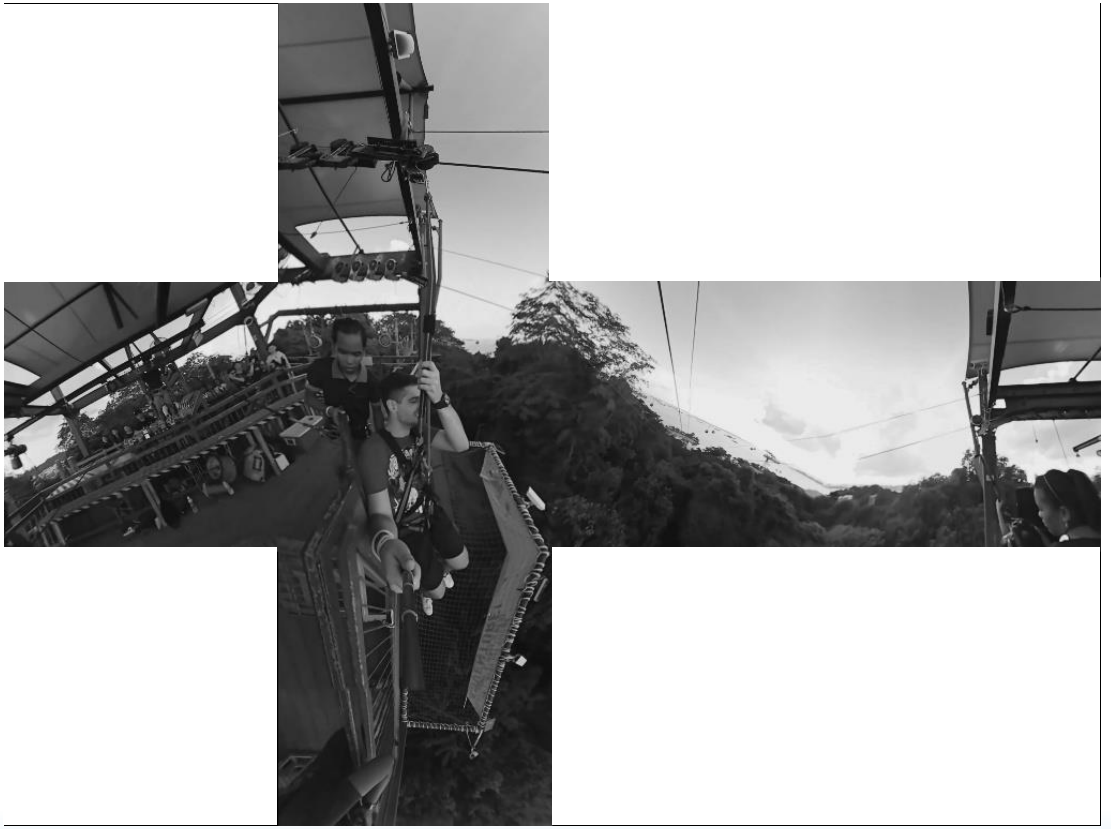
Outline

- 360° Video Hardware
- Layout Projection
- Optimizing the Streaming Delivery of 360° Videos
- Viewing Direction Determination
- Video Stabilization
- Salient Object Identification

Equirectangular Projection Layout

- Equirectangular layout is used by 360 video cameras
- Significant distortions occur at the “polar regions”





Example: Sentosa *MegaZip* (SG)

Example: Marina Bay (SG)

“Magic Window” (Narrow FOV)

Challenges (1)

- Projection Layouts
  - There is no distortion free projection from a sphere to a planar layout. Hence, various projection layouts have been proposed with different properties:
    - Equirectangular
    - Cube Map
    - Pyramid
    - Offset Cube Map
  - Optimizing the Delivery of 360-Degree Videos in Bandwidth-Constrained Networks

Challenges (2)

- Video Stabilization
  - The consumer 360 video cameras are small and light and it is difficult to keep them steady.

- Determine the Viewing Direction
  - While it is fun to explore the viewing direction by manually panning left, right, up and down, it can also be distracting.
  - It is possible to miss important objects or events, because they occur “behind” the viewer.
  - Finding the motion direction; projecting such that lines are straight, etc.

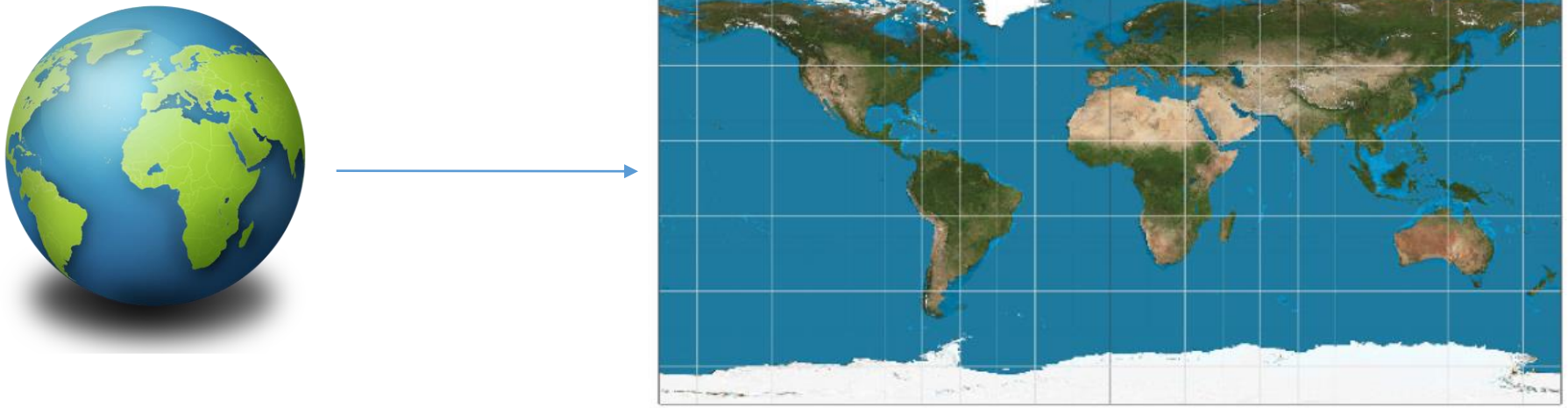
Optimizing the Delivery of 360-Degree Videos in Bandwidth-Constrained Networks

Bayan Ta'ani
bayan@comp.nus.edu.sg

360-Degree Videos – An Overview

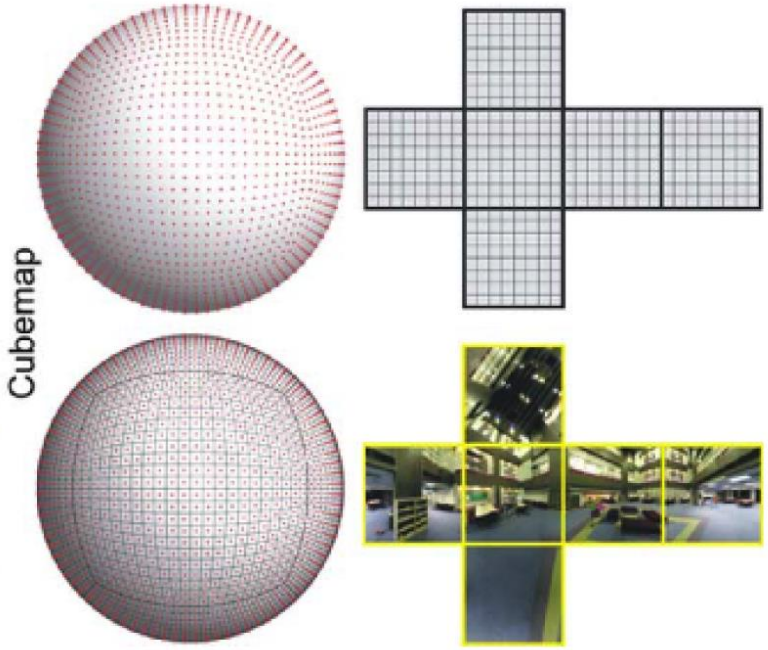
Representing Spherical Video in 2D

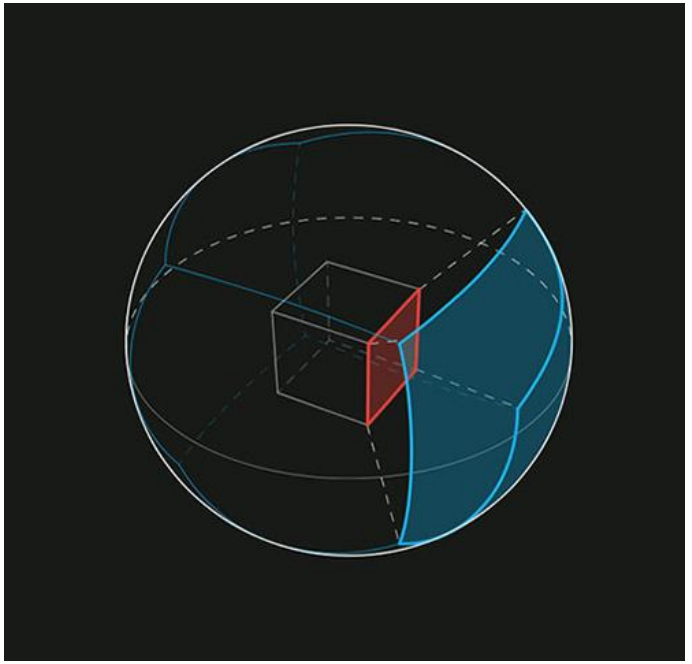
- Equirectangular Projection: unwraps a sphere on a 2D rectangular plane with dimensions $2\pi r \times \pi r$.
  - Has many redundant pixels, especially around the poles.
  - Used by YouTube.



360-Degree Videos – An Overview

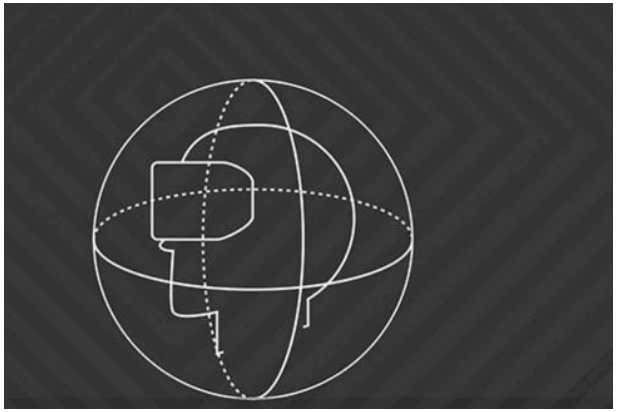
- Cube Map Projection: projects a sphere on a cube, then unwraps the cube.
  - No distortion at poles, they are treated the same as equator areas.
  - Reduces file size by 25% against equirectangular.
  - Used by Facebook.



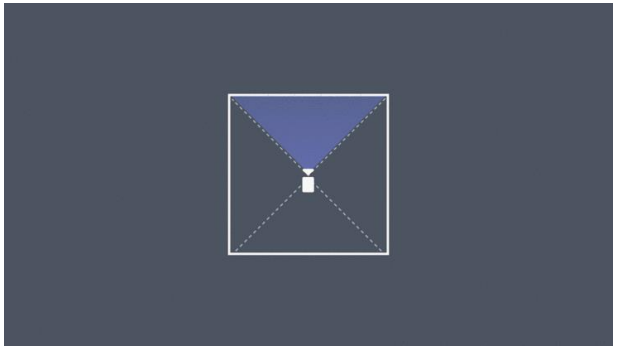


360-Degree Videos – An Overview

- Pyramid Projection: project sphere on pyramid, where the base is the front view of the user.
  - Only the viewport is rendered in full resolution.
  - Reduces file size by 80% against equirectangular.
  - To handle different field-of-views (FoV), Facebook creates 30 different versions of the same video covering most of the FoVs (separated by ~30°).



- Offset Cube Map:
  - Built on top of cube map.
  - Designed to overcome the problems of pyramid projection (GPU support).
  - Still suffers from the same storage problem.



Outline

- Optimizing the Delivery of 360-Degree Videos under Bandwidth-Constrained Networks
  - Overview of 360-Degree Videos
  - Problem Statement
  - Related Work
  - Proposed Solution – Scalable, Tile-Based Viewport-Adaptive 360 Video Streaming
  - Conclusions

Problem Statement (1)

360° video file sizes are very large

- E.g., Facebook's **Surround 360** outputs 4K, 6K or 8K per eye, with frame rate of 30 or 60 fps[1]
- Unlike non-360 videos, this number represents the quality of the whole frame, the rendered part is actually much less than that.
  - FoV is 110° horizontally and 90° vertically of the entire sphere.

High bandwidth consumption!

- Bandwidth is wasted on the unwatched parts of the 360 view (wastes ~80% of the bandwidth (Qian et al., 2016))

[1] https://code.facebook.com/posts/1755691291326688/introducing-facebook-surround-360-an-open-high-quality-3d-360-video-capture-system/

Problem Statement (2)

Stream the user Field-of-View (FoV)

- Generate multiple versions of each frame for possible FoVs.
- Stream only FoV in high quality, the rest in low quality.
  - Achieved using tiling

- Computationally expensive
- More delay
- The need of multiple decoders at client to play out tiles of different qualities

Outline

- Optimizing the Delivery of 360-Degree Videos under Bandwidth-Constrained Networks
  - Overview of 360-Degree Videos
  - Problem Statement
  - Related Work
  - Proposed Solution – Scalable, Tile-Based Viewport-Adaptive 360 Video Streaming
  - Conclusions

Related Work

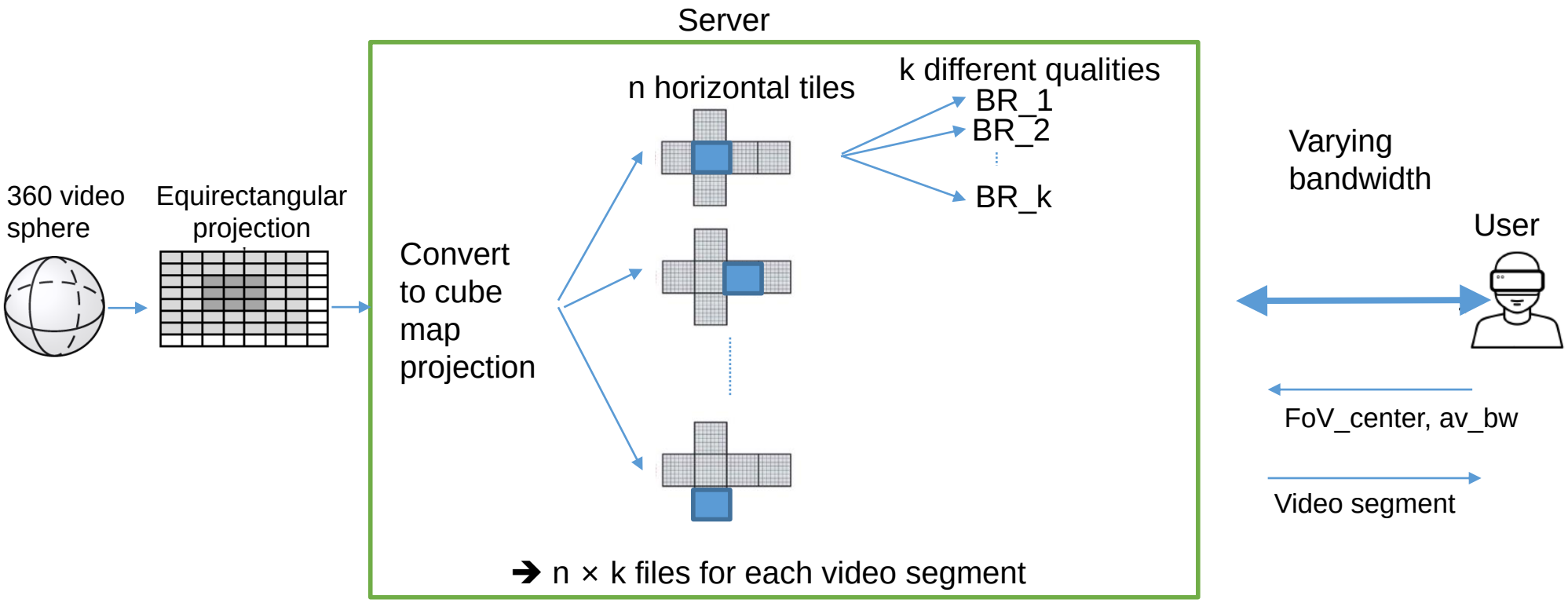
Tiling schemes:

- (El-Ganainy, 2017) develop a tile segmentation scheme for 360 videos using cube map projection based on the nature of capturing and viewing immersive sports videos called *tiled_cubemap*, and formulate a rate adaptation utilizing *tiled_cubemap*
- (Zare et al., 2016) generate 2 versions of the video at different resolutions, each divided into multiple tiles using HEVC
- (Graf et al., 2017) propose 3 tiling strategies (*full delivery basic, full delivery advanced, partial delivery*) and test it in an Android and Web environment
- (Hosseini et al. 2016) used tiling of 360-degree video frames and integrate the method into MPEG-DASH SRD, and using this tiling, they adapted to the viewport of the client.

Related Work

- Corbillon et al. [3]
  - study the impact of the number of Quality Emphasis Centers (QEC), the video segment length, and the video projection method.
  - Their results show that the CubeMap method results in the best quality as measured by MS-SSIM compared to Equirectangular, Pyramid and Dodecahedron projection methods
  - Segment duration of two seconds provides the best trade-off between encoding complexity and re-synchronization with the head movement and fast adaptation.
- Qian et al.,
  - Analyzing user head movement data to predict the next viewport.
- Rondao et al. [15]
  - Predicted head movements and used tiling at the server side to stream the tiles that overlapped with the predicted viewport.
- Hosseini et al. [8, 9]
  - used tiling of 360-degree video frames and integrate the method into MPEG-DASH Spatial Relation Description (SRD) [13], and using this tiling, they adapted to the viewport of the client.
- Yu et al. [20]
  - formulated an optimization algorithm for the adaptation of 360-degree video representations by jointly optimizing the sampling and coding stages. The sampling distortion represents the redundant pixels resulting from projection techniques, while the coding distortion is the one introduced by the video encoder in order to reduce the bitrate.

Example: (Corbillon et al., 2016):

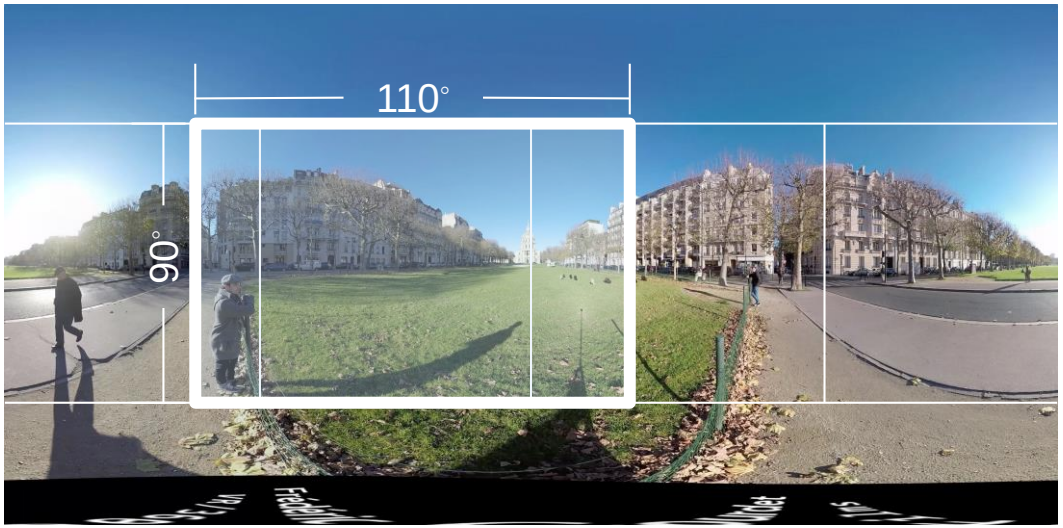


Outline

- Optimizing the Delivery of 360-Degree Videos under Bandwidth-Constrained Networks
  - Overview of 360-Degree Videos
  - Problem Statement
  - Related Work
  - Proposed Solution – Scalable, Tile-Based Viewport-Adaptive 360 Video Streaming
  - Conclusions

Scalable, Tile-Based Viewport-Adaptive 360 Video Streaming

- Tiling

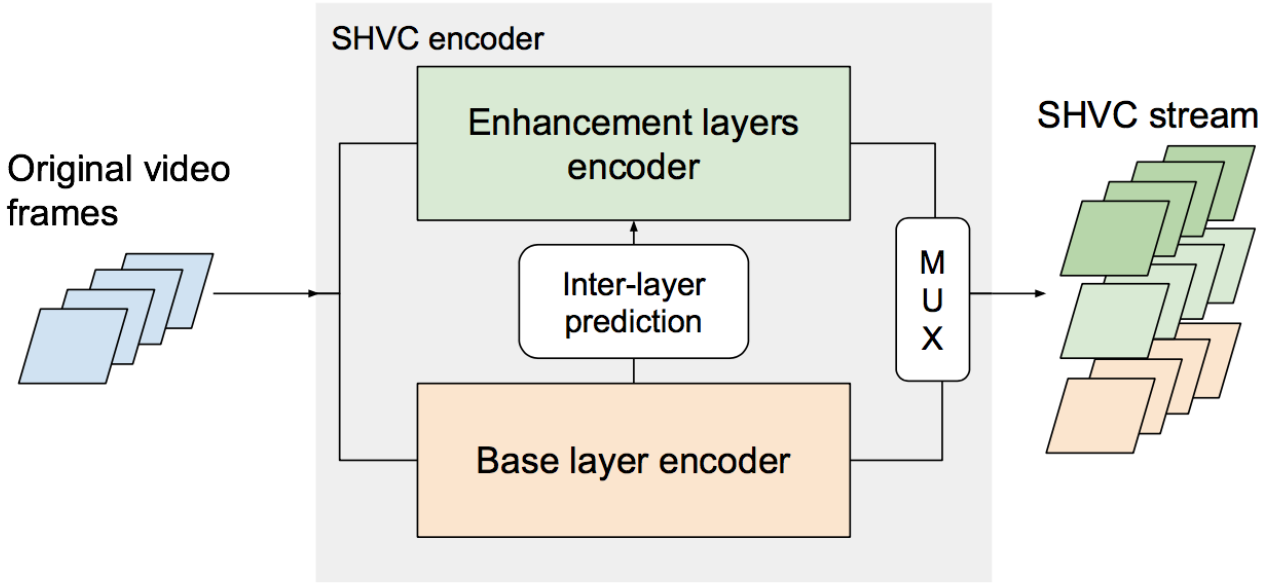


Drawbacks:

- Increase in the number of generated files, increased encoding time, complexity of the manifest files.

Scalable, Tile-Based Viewport-Adaptive 360 Video Streaming

- Scalable Coding

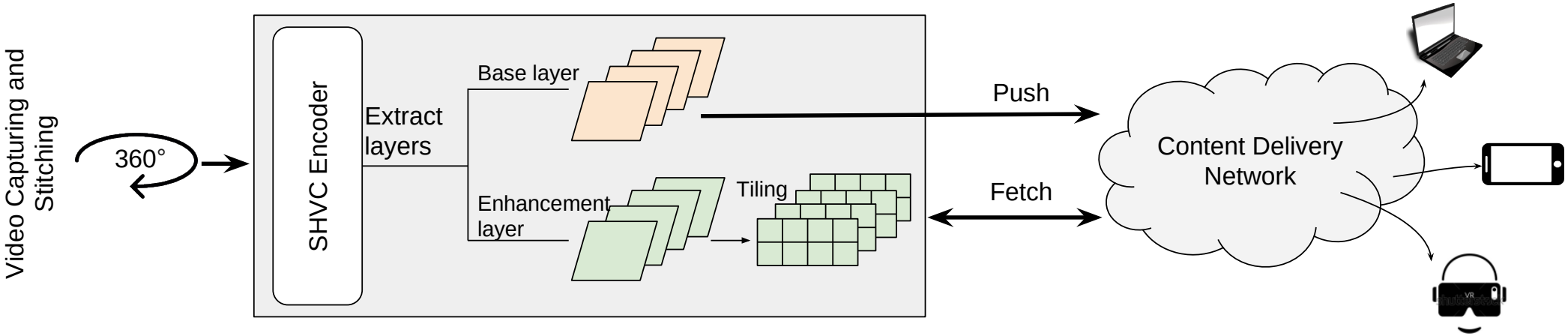


- In the worst network conditions the lowest quality layer will be downloaded (stalls will be eliminated).
- This is useful for interactive applications like 360-degree video streaming, where the bandwidth requirements are more stringent due to the increased size of the videos.

Scalable, Tile-Based Viewport-Adaptive 360 Video Streaming

High-quality viewport plus lowest-quality background

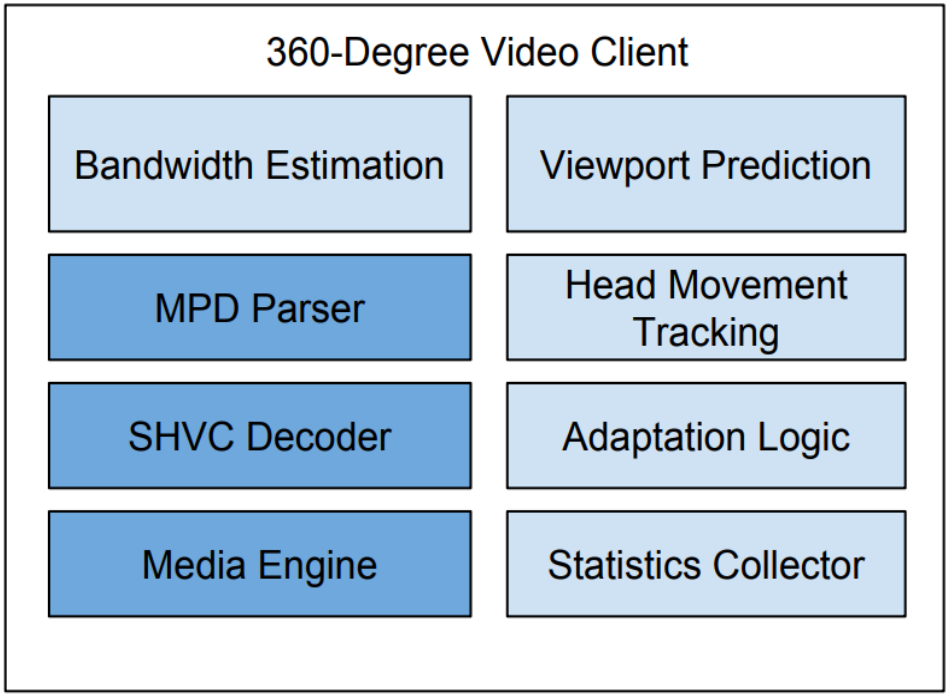
- Server receives original video in a panoramic view
- transcoding and tiling is performed to support all viewport and network adaptations
- Client detects the viewport and requests the tiles that overlap with the viewport in high quality, while the background is transmitted by the server in the lowest quality.



Scalable, Tile-Based Viewport-Adaptive 360 Video Streaming

360-Degree Video Player

1. Viewport prediction of the next segment
2. Quality decision of the next segment based on the estimated bandwidth
3. Delivery:
   - Low bandwidth ➔ nothing additional is requested from the server (the BL is still pushed by the server)
   - High bandwidth enough to accommodate the EL tiles ➔ appropriate tiles from the EL are requested from the server.
4. Video playback



Outline

- Optimizing the Delivery of 360-Degree Videos under Bandwidth-Constrained Networks
  - Overview of 360-Degree Videos
  - Problem Statement
  - Related Work
  - Proposed Solution – Scalable, Tile-Based Viewport-Adaptive 360 Video Streaming
  - Conclusions

Discussion

- By deploying SHVC, the scalable extension of the state-of-the- art video codec HEVC, we minimize the number of requests for the base layer by pushing it to the client during both playback and buffering states.
- To minimize the bandwidth waste that results from transmitting parts of the panoramic view that are not watched by the user, we leverage head movement patterns to predict the viewport of the next segment, and use tiling of the enhancement layers at the server side to provide high-quality viewport adaptation.

References

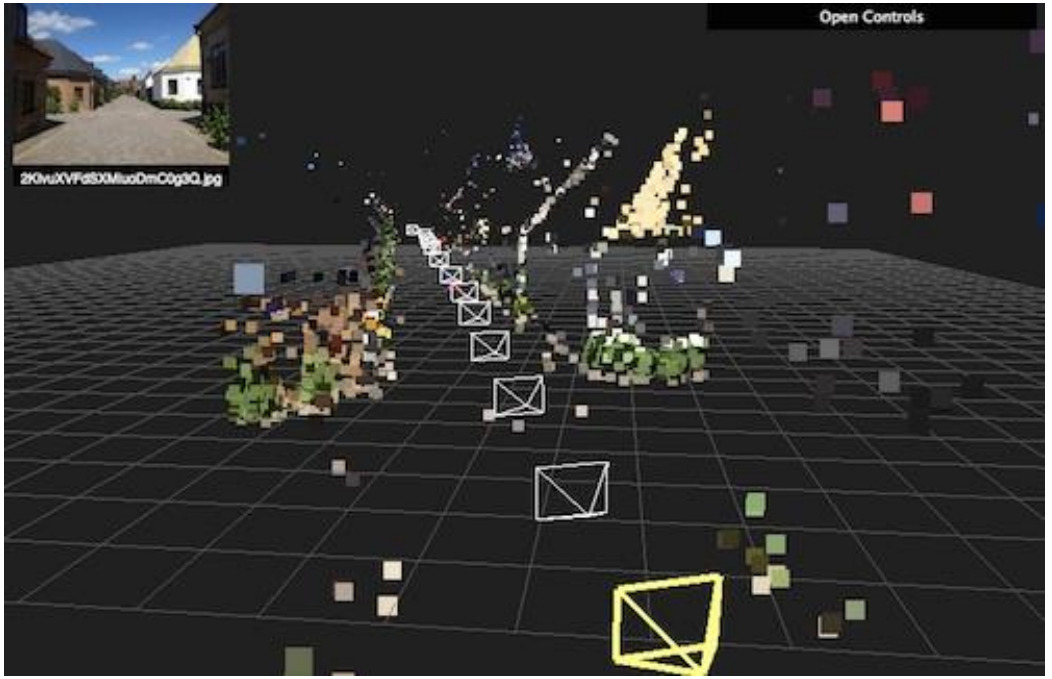
- F. Qian, L. Ji, B. Han, and V. Gopalakrishnan. Optimizing 360 video delivery over cellular networks. In Proceedings of the 5th Workshop on All Things Cellular: Operations, Applications and Challenges, pages 1–6. ACM, 2016.
- Gaddam, Vamsidhar Reddy, et al. "Tiling in interactive panoramic video: Approaches and evaluation." *IEEE Transactions on Multimedia* 18.9 (2016): 1819-1831.
- Graf, Mario, Christian Timmerer, and Christopher Mueller. "Towards Bandwidth Efficient Adaptive Streaming of Omnidirectional Video over HTTP: Design, Implementation, and Evaluation." *Proceedings of the 8th ACM on Multimedia Systems Conference*. ACM, 2017.
- El-Ganainy, Tarek. "Spatiotemporal Rate Adaptive Tiled Scheme for 360 Sports Events." *arXiv preprint arXiv:1705.04911* (2017).
- Zare, Alireza, et al. "HEVC-compliant tile-based streaming of panoramic video for virtual reality applications." *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016.
- Hosseini, Mohammad, and Viswanathan Swaminathan. "Adaptive 360 VR video streaming: Divide and conquer." *Multimedia (ISM), 2016 IEEE International Symposium on*. IEEE, 2016.
- P. Rondao Alface, J.-F. Macq, and N. Verzijp. Interactive omnidirectional video delivery: A bandwidth-effective approach. *Bell Labs Technical Journal*, 16(4):135–147, 2012.
- M. Yu, H. Lakshman, and B. Girod. Content adaptive representations of omnidirectional videos for cinematic virtual reality. In *Proceedings of the 3rd International Workshop on Immersive Media Experiences*, pages 1–6. ACM, 2015.

Outline

- 360° Video Hardware
- Layout Projection
- Optimizing the Streaming Delivery of 360° Videos
- Viewing Direction Determination
- Video Stabilization
- Salient Object Identification

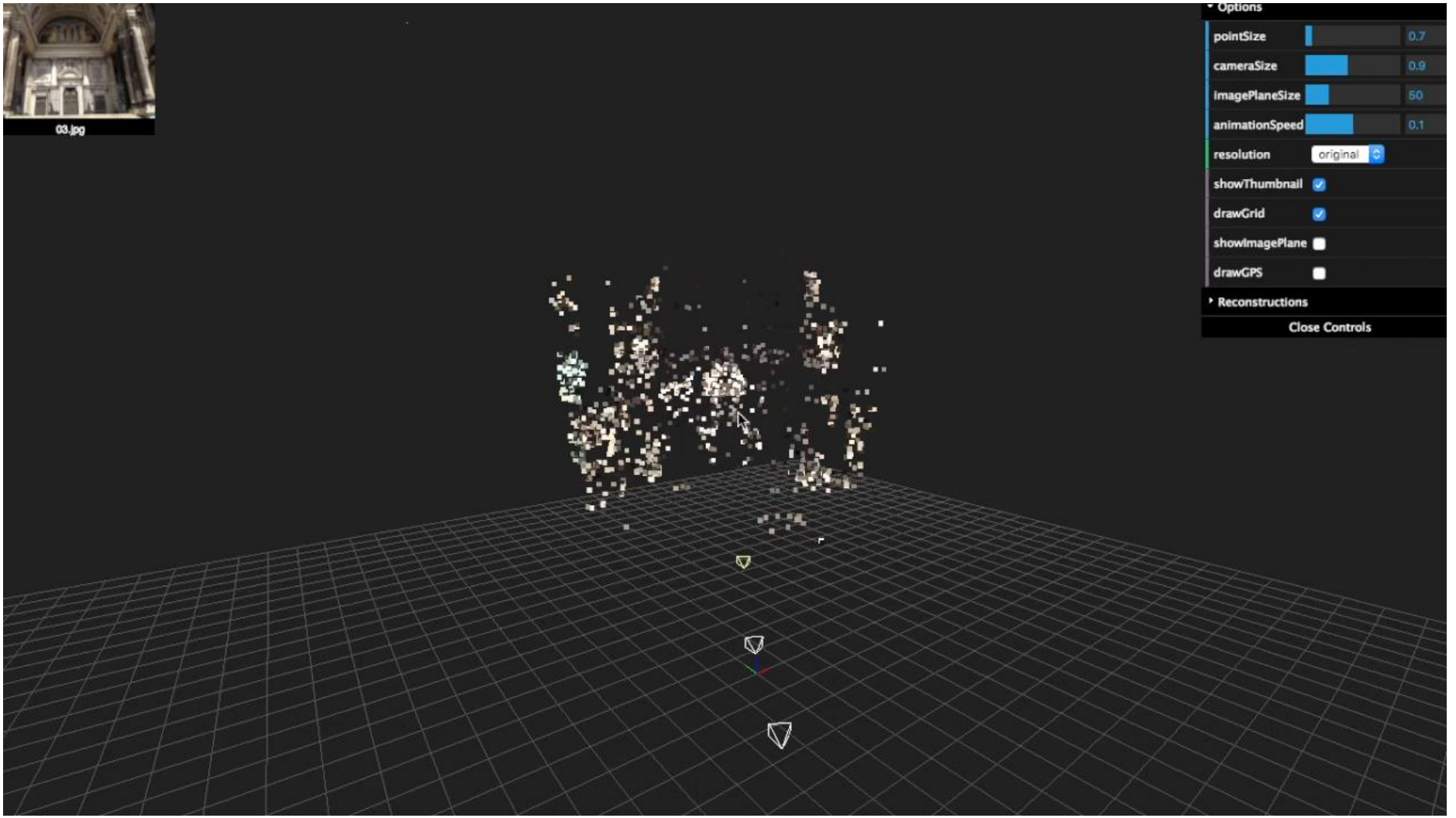
OpenSfM

- Structure from Motion (SfM) library written in Python on top of OpenCV.
- Estimation of three-dimensional structures from two-dimensional image sequences that may be coupled with local motion signals.
- Estimation/reconstruction of:
  - Camera poses
  - 3D structures/scenes
- Performance is often slow (pair-wise image matching through SIFT features, bundle adjustment)

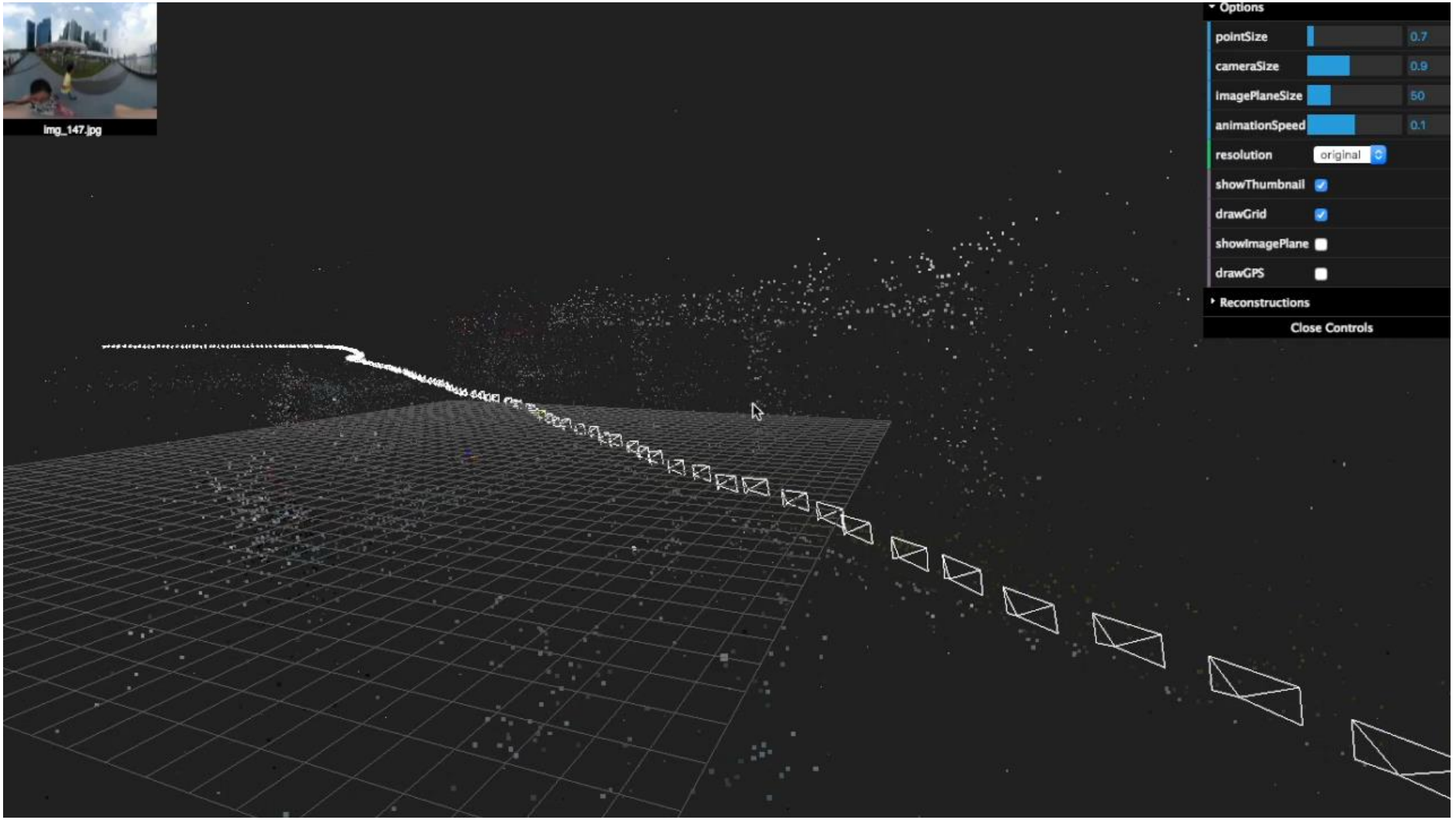


© OpenSfM

OpenSfM Example: 3D



OpenSfM Example: Camera Pose



OpenSfM Performance

Performance: 45-second video at 3fps

- opensfm (AMD Opteron 4184; 2.8 GHz; 4 cores of 6-core CPU; 4 processes): 43:57 minutes; 353%
- cat09 (2 x Intel E5-2630 v2; 2.6 GHz; 6 cores each; 15 processes): 11:13 minutes; 1120% CPU
- elk01 (2 x Intel E5-2630 v3; 2.4 GHz; 8 cores each; 20 processes): 10:22 minutes; 1254% CPU

- http://matterhorn.d2.comp.nus.edu.sg/~rzimmerm/viewer/reconstruction.html#file=data/marinabay/reconstruction.meshed.json

Outline

- 360° Video Hardware
- Layout Projection
- Optimizing the Streaming Delivery of 360° Videos
- Viewing Direction Determination
- Video Stabilization
- Salient Object Identification

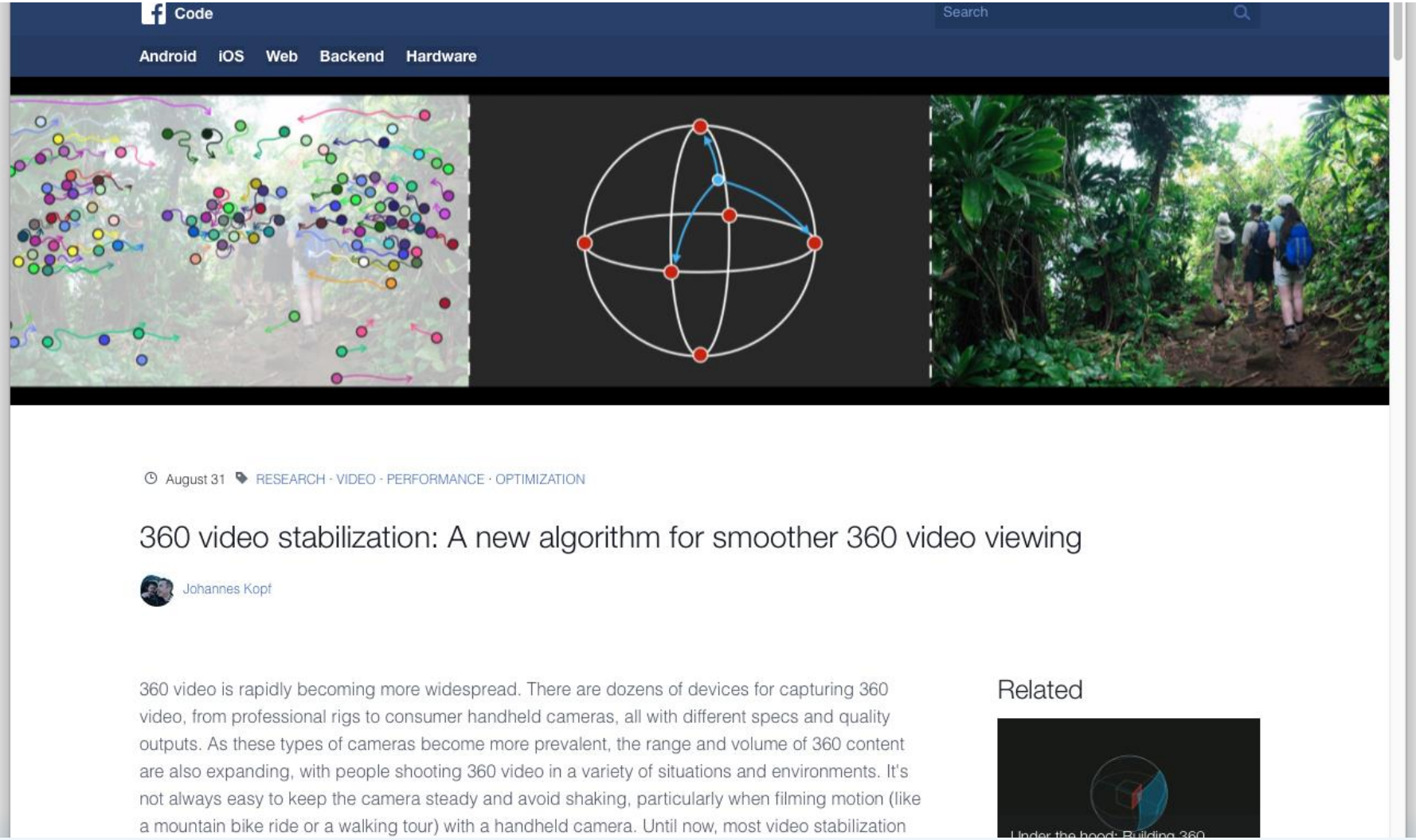
Stabilization

- Narrow FOV: 3 Steps
  - Tracking of motion vectors
  - Fitting of motion model (e.g., 2D: Trans + Rot + Scale) + smoothing
  - Cropping

- Cropping is not needed for 360 video
- 3D structure information can be useful

Stabilization: Johannes Kopf (FB)

- Stabilization that uses both 2D and 3D information



Stabilization: Johannes Kopf (FB)

360 Video Players

- Self-hosted HTML5 Open Source 360-Degree Video Player
  - https://github.com/littlstar/axis
- Valiant 360
  - http://flimshaw.github.io/Valiant360/
- Google VR View
  - https://developers.google.com/vr/concepts/vrview
- YouTube
  - Recognizes 360 videos during upload.
- Facebook
  - Now also supports 360 videos.

Outline

- 360° Video Hardware
- Layout Projection
- Optimizing the Streaming Delivery of 360° Videos
- Viewing Direction Determination
- Video Stabilization
- Salient Object Identification

Salient Object Identification

I. Object Proposal Detection

II. Object Ranking

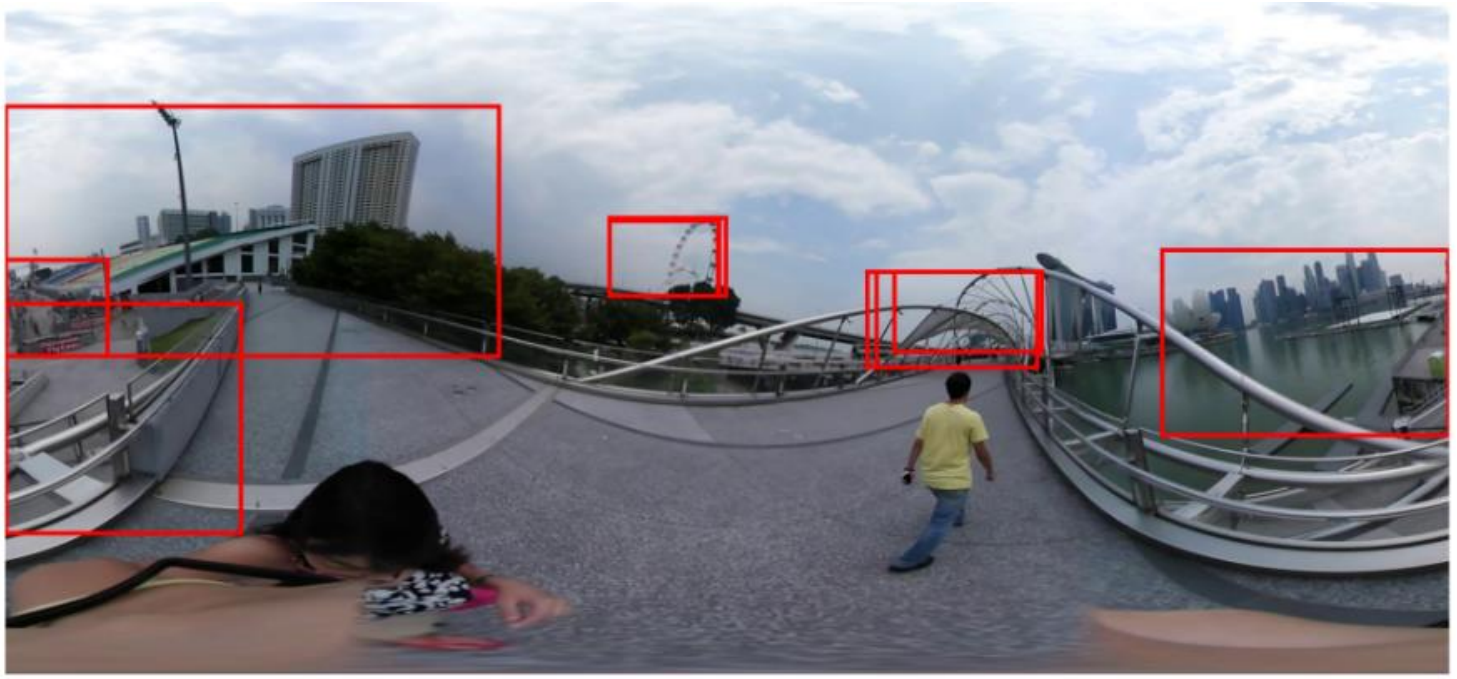
III. Viewing Path Rendering

IV. Examples

Object Proposal Detection (1)

Detect possible objects of interest in every frame

- Selective Search algorithm [1]
- Edge Boxes [2]



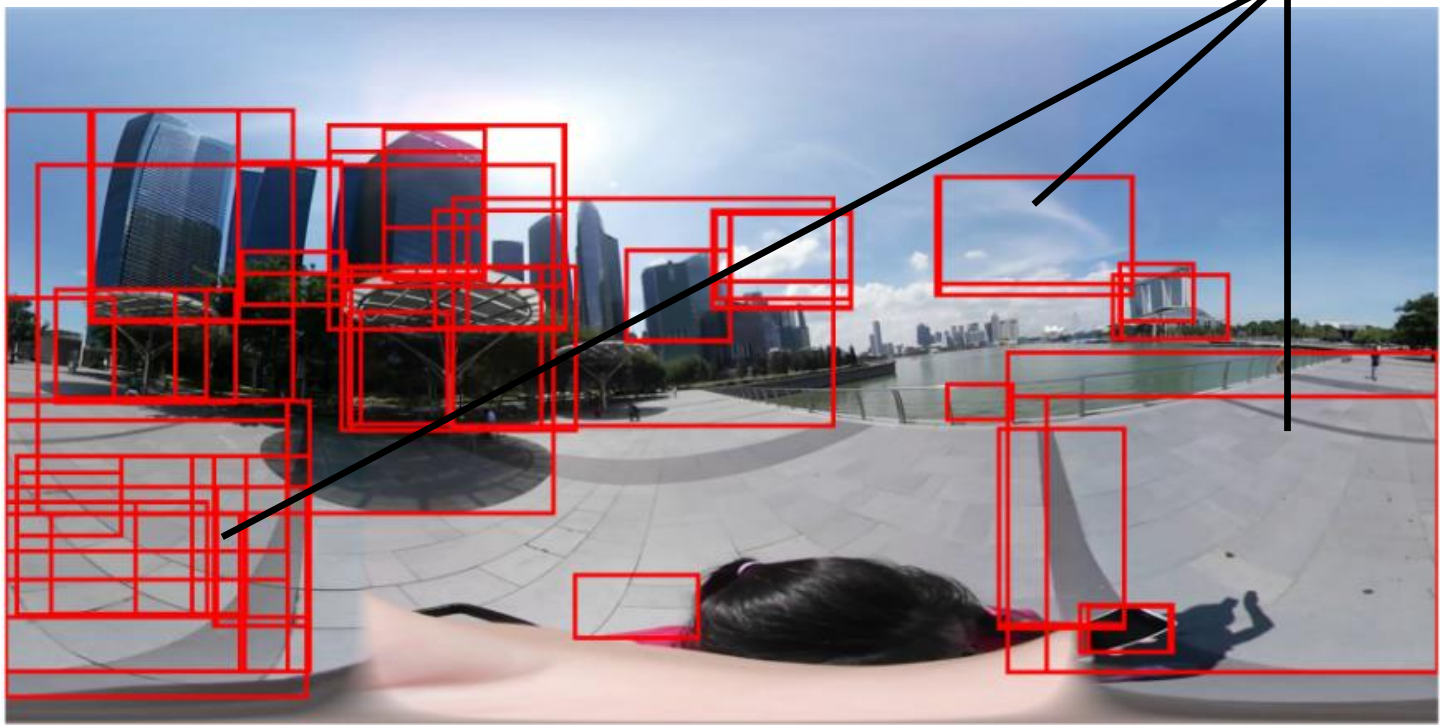
[1] Uijlings, Jasper RR, et al. "Selective search for object recognition." International journal of computer vision 104.2, 2013.

[2] Zitnick, C. Lawrence, and Piotr Dollár. "Edge boxes: Locating object proposals from edges." European Conference on Computer Vision. Springer International Publishing, 2014.

Object Proposal Detection (2)

Detect possible objects of interest in every frame

- Selective Search algorithm [1]
- Edge Boxes [2]



[1] Uijlings, Jasper RR, et al. "Selective search for object recognition." International journal of computer vision 104.2, 2013.

[2] Zitnick, C. Lawrence, and Piotr Dollár. "Edge boxes: Locating object proposals from edges." European Conference on Computer Vision. Springer International Publishing, 2014.

Object Proposal Detection

Negative Object Proposal Removal

- Object proposals detected may belong to the background and contain no objects at all, and should be removed.

- Pre-trained deep models can be used to detect objects from certain classes and remove object proposals that do not belong to any of the classes (with a low detection score).
  - Places205-AlexNet: 205 scene categories
  - Hybrid-CNN: 1183 categories (205 scene categories and 978 object categories)
  - Places205-GoogLeNet: 205 scene categories

[These 3 models are pre-trained on ImageNet, and come with the *Caffe* deep learning framework.]

https://github.com/BVLC/caffe/wiki/Model-Zoo

Salient Object Identification

I. Object Proposal Detection

II. Object Ranking

III. Viewing Path Rendering

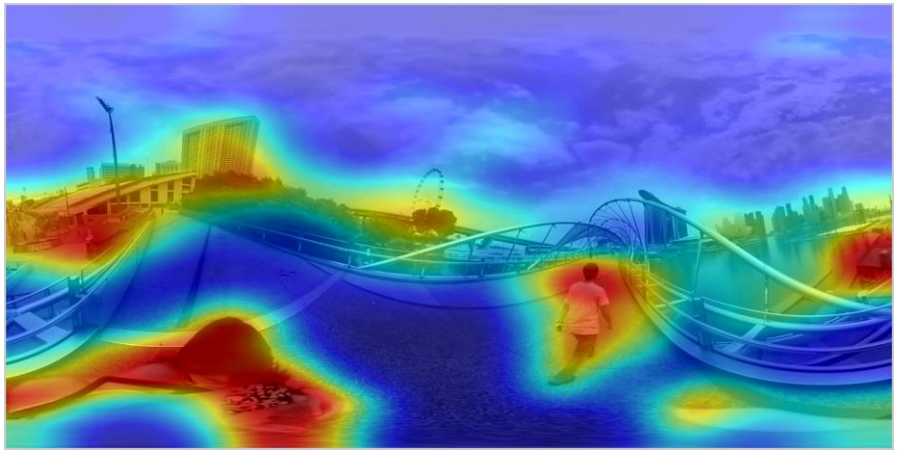
IV. Examples

Object Ranking (1)

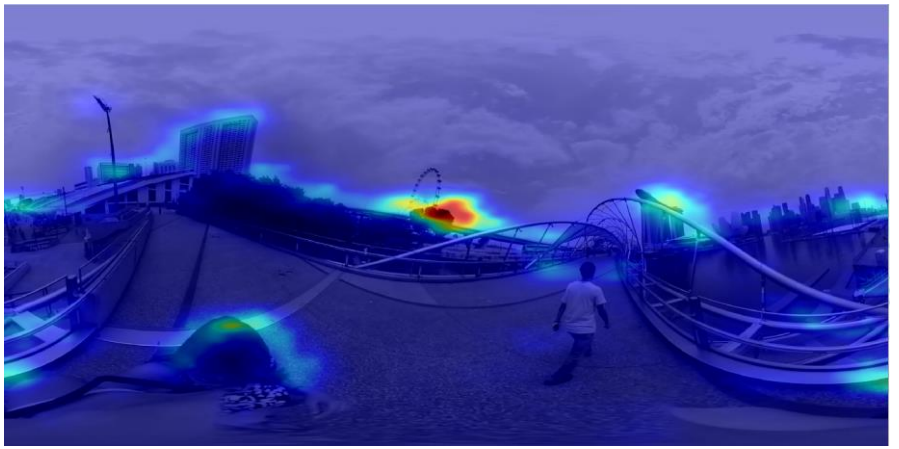
The saliency of objects can be estimated based on:

- Content-based Saliency Detection

[1] Static and space-time visual saliency detection by self-resemblance, Hae Jong Seo, Peyman Milanfar, Journal of Vision (2009) 9(12):15, 1–27



Static Saliency Map

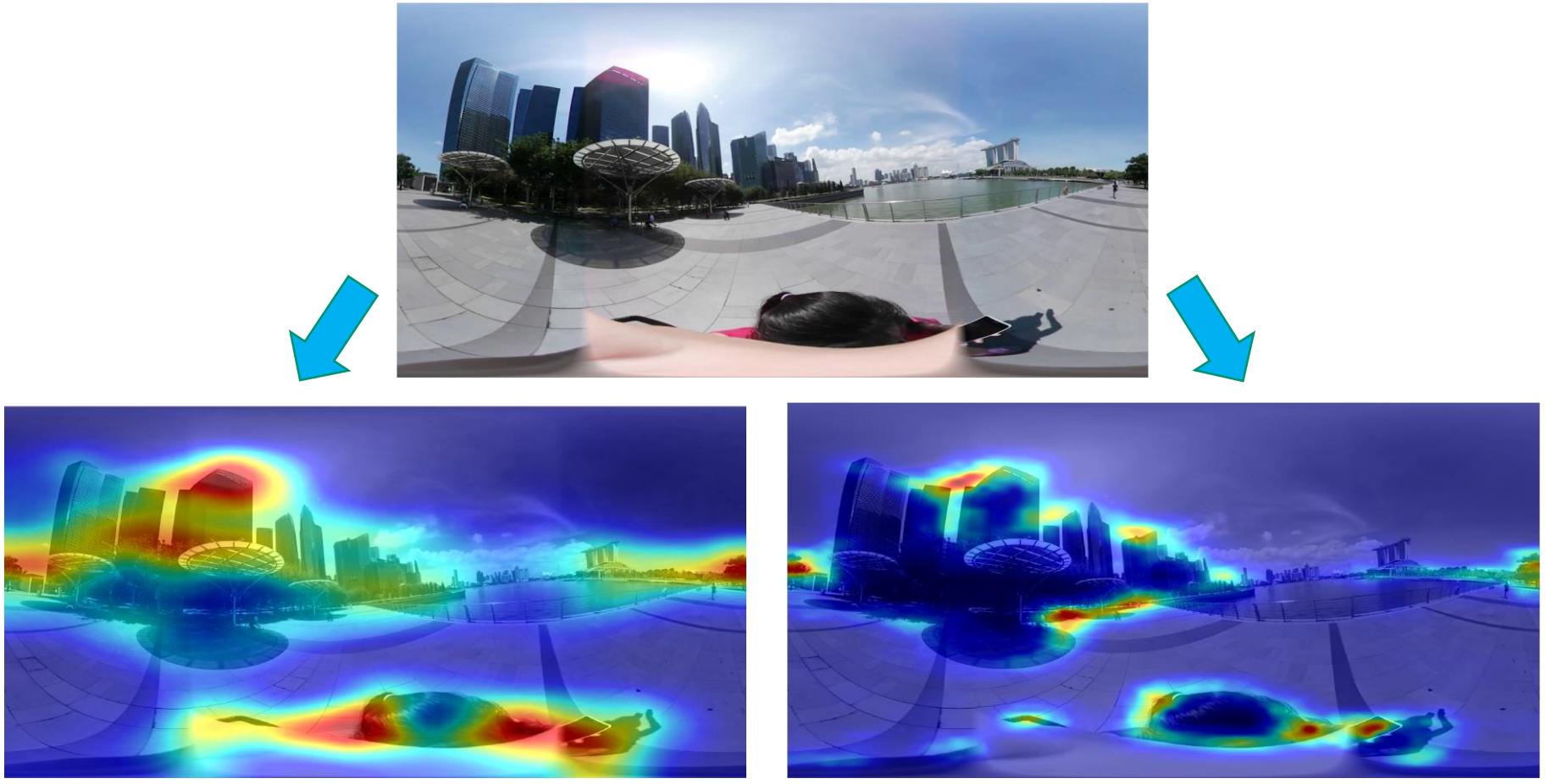


Space-time Saliency Map

Object Ranking (2)

The saliency of objects can be estimated based on:

- Content-based Saliency Detection



Static Saliency Map

Space-time Saliency Map

Object Ranking (3)

The saliency of objects can be estimated based on:

- User Preference
  - Classify the objects into a list of pre-defined categories of cognitive content and/or affective content.
    - Cognitive content: object, scene, or event categories.
    - Affective content: low, high, normal emotion intensities.
  - Choose the preference of camera motion: still, panning, tracking…

[1] Personalized Video Adaptation Based on Video Content Analysis, Min Xu, Jesse S. Jin, Suhuai Luo, MDM/KDD 2008

- Video Quality
  - Distortion,
  - Occlusion,
  - Illumination,
  - and others

Salient Object Identification

I. Object Proposal Detection

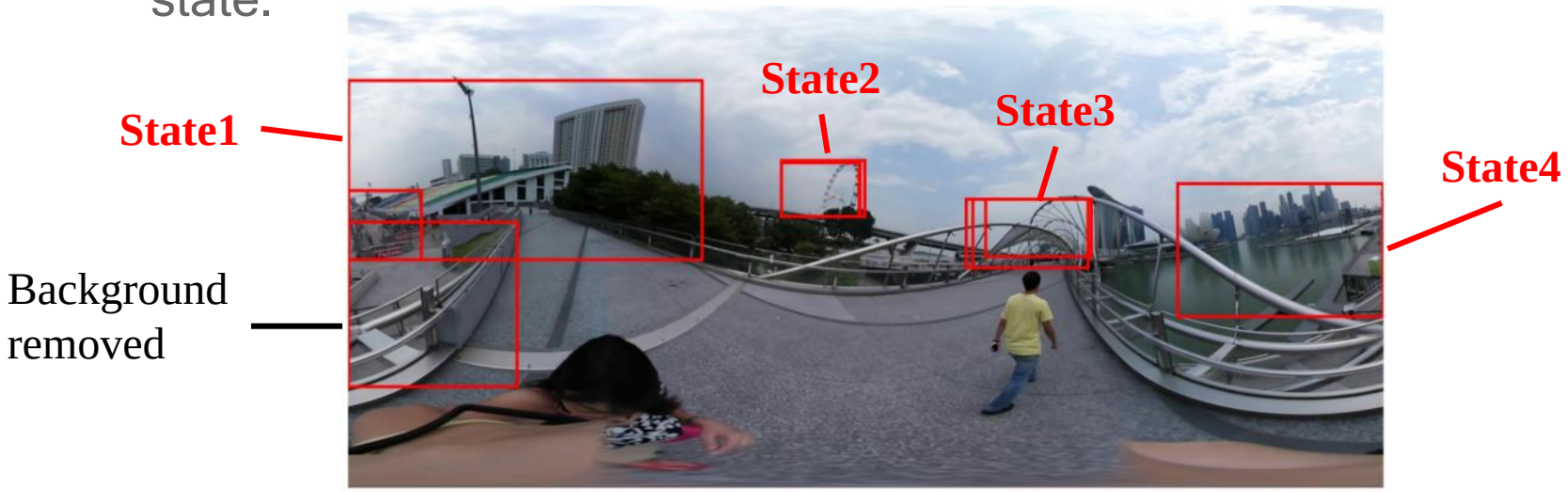
II. Object Ranking

III. Viewing Path Rendering

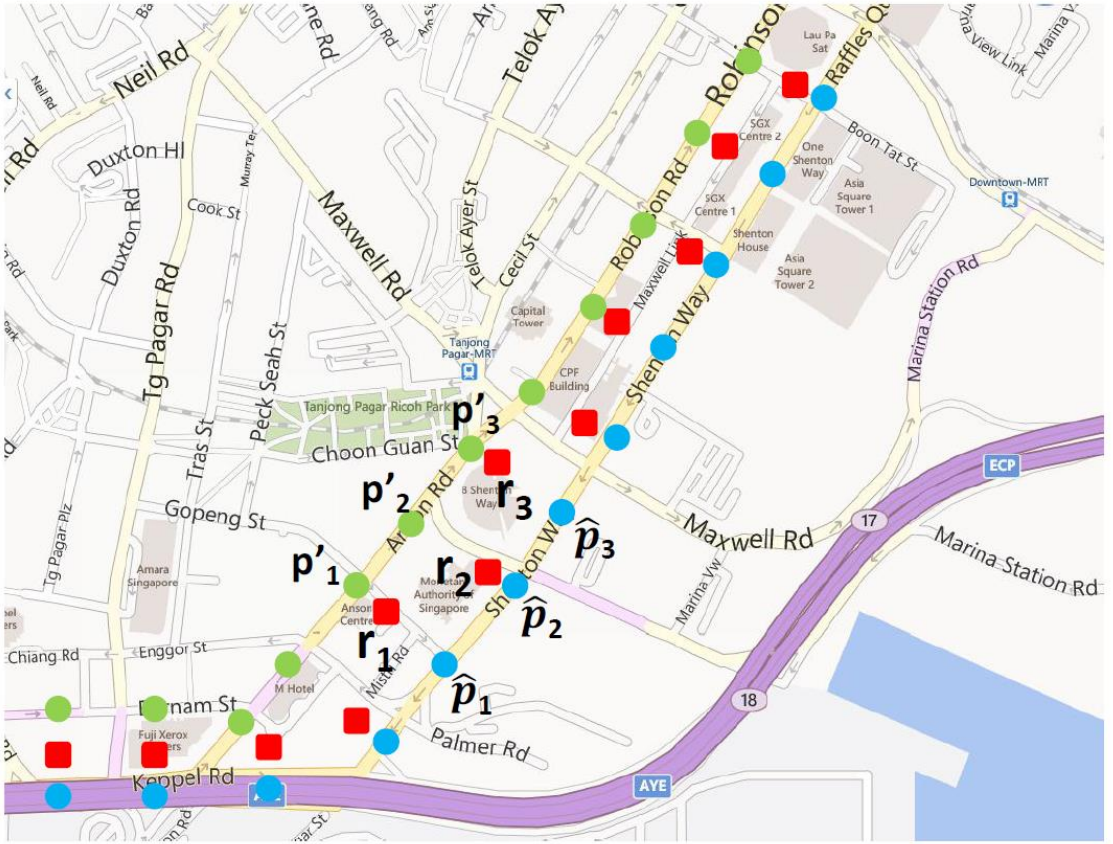
IV. Examples

Viewing Path Rendering (1)

- The rendering of salient objects can be learned from user-generated videos
  - Hidden Markov Model (HMM)
    - States: salient objects to render.
    - Transition probabilities: the transition probability between states.
    - Emission probabilities: the distribution of shot length on each state.



- HMM is used in map matching to find the most likely path, given a sequence of noisy location points.



Viewing Path Rendering (2)

- Identify the objects of each state from user-generated videos.
- Estimate the transition and emission matrices.

	State1	State2	State3	State4
State1	0	0.6	0.1	0.3
State2	0.2	0	0.6	0.2
State3	0.05	0.45	0	0.5
State4	0.2	0.1	0.7	0

Transition Matrix

	10 sec	20 sec	30 sec	40 sec	50 sec
State1	0.8	0.2	0	0	0
State2	0.3	0.6	0.1	0	0
State3	0.05	0.25	0.4	0.2	0.1
State4	0.1	0.3	0.35	0.2	0.05

Emission Matrix

- An example video shot moving from State2 to State3:

http://api.geovid.org/v1.0/gv/video/8d1aa928a3161a13e4067778cf802944e04a6966/high

Crowd-sourced Videos

Salient Object Identification

I. Object Proposal Detection

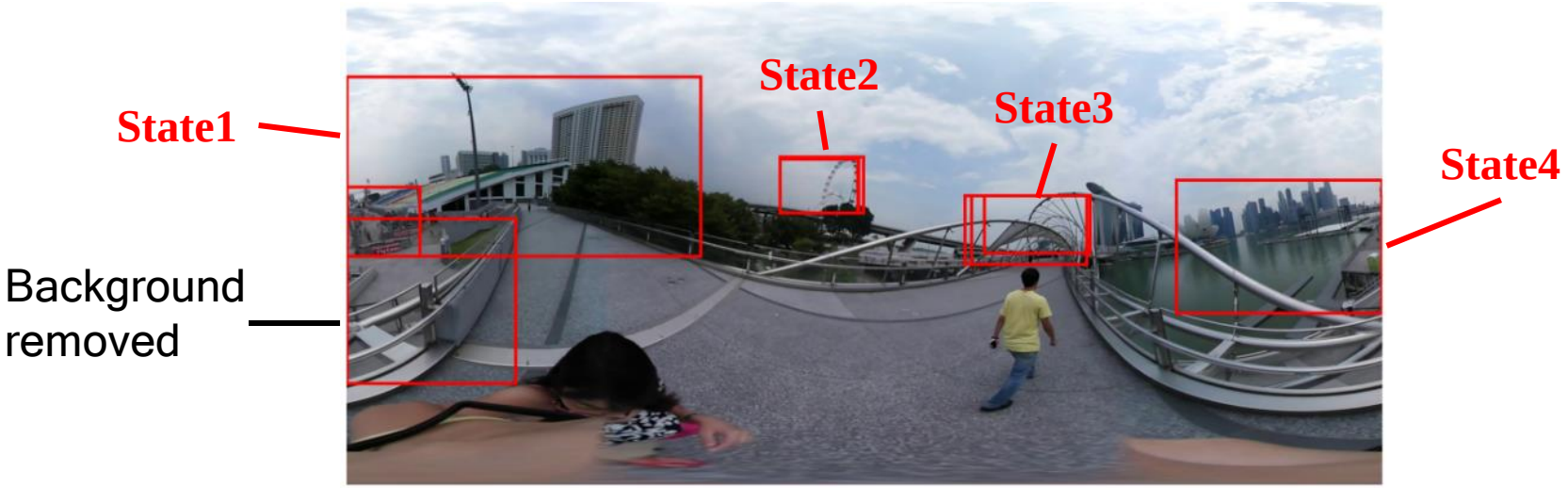
II. Object Ranking

III. Viewing Path Rendering

IV. Examples

Examples (1)

- Scenario: user prefers famous landmarks or still camera motion.



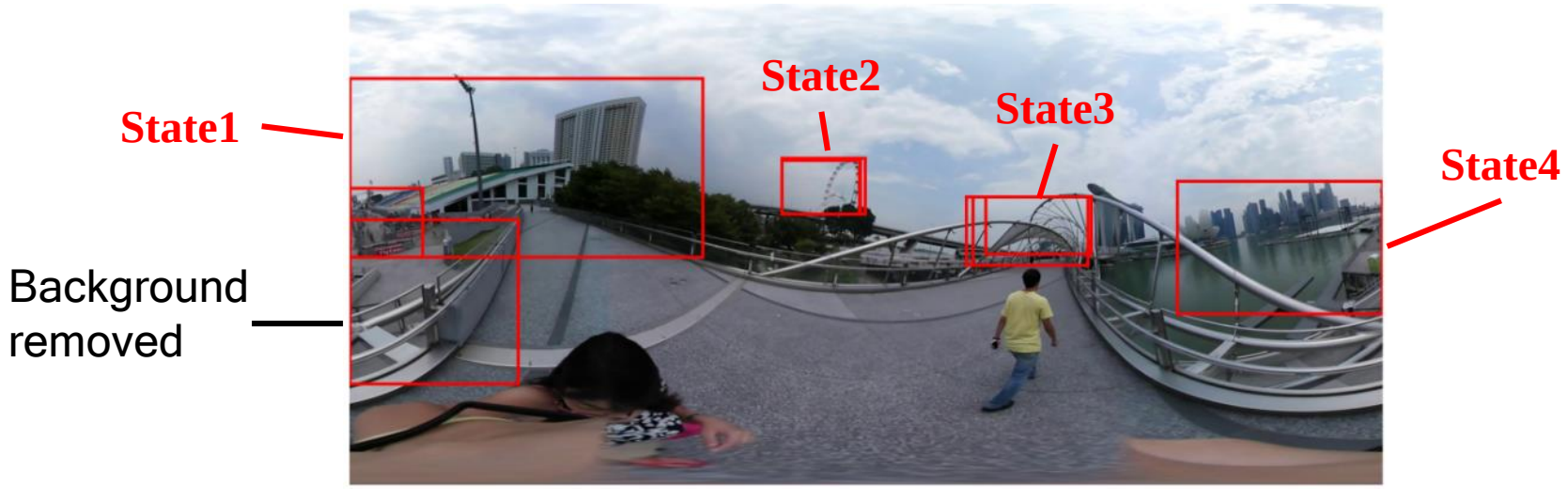
Possible rendering route:

State2 ➔ State3

- Based on the transition matrix, the transition probability from State2 to State3 (0.6) is larger than the transition probability from State3 to State2 (0.45).

Examples (2)

- Scenario: user wants to see all the salient objects.



Possible rendering route:

State1 ➔ State2 ➔ State3 ➔ State4

- State2: Singapore Flyer
- State3: Helix Bridge + Marina Bay Sands hotel
- State4: Financial Centre

More Challenges

- Robust saliency detection
- Salient object ranking
- Viewing path determination and optimization
- …

Summary

- 360° video technology is an interesting area with many new challenges.
- Fundamentally 360° videos require a tremendous amount of resources (e.g., bandwidth, storage).
- Some of the existing techniques can be adapted, for example MPEG-DASH, but many parts need to be optimized for a high quality user experience.

- Much interesting work still needs to be done!


Q
&
A


danke
рахмат
謝謝
ngiyabonga
teşekkür ederim
Баярлалаа
спасибо
faafetai lava
mersi
kia ora
barka
welalin
tack
spas
vinaka
спасибі
blagodaram
dank je
misaotra
matondo
paldies
grazzi
mahalo
tapadh leat
kiitos
dankie
nanni
nandri
dhanyavad
thank you
gracias
хвала
asante
manana
bedankt
mauruuru
köszönöm
akun
dankon
aciu
obrigada
tenki
murakoze
enkosi
bayarlalaa
gràçie
hvala
djiere dieuf
tau
mochchakkeram
дякую
mamnun
chokrane
dziękuję
chnorakaloutioun
gratias ago
gràcies
sulpáy
go raibh maith agat
sobodi
děkuji
obrigado
sagolun
sukriya
kop khun krap
taiku
arigatô
takk
dakujem
trugarez
mèsi
didi madloba
najis tuke
ありがとう
tanemirt
grazie
kam sah hamnida
rahmat
terima kasih
rahmet
ευχαριστώ
diolch
dhanyavadagalu
shukriya
mercé
мерси
তোমাকে ধন্যবাদ
감사합니다
xièxie
merci