A virtual tour of free viewpoint rendering

Cédric Verleysen

ICTEAM institute, Université catholique de Louvain, Belgium

cedric.verleysen@uclouvain.be

Organization of the presentation

- Context
- Acquisition setups
- Rendering virtual views from images
- Rendering virtual views from 3D model
- Image-based rendering vs model-based rendering

Organization of the presentation

- Context
- Acquisition setups
- Rendering virtual views from images
- Rendering virtual views from 3D model
- Image-based rendering vs model-based rendering

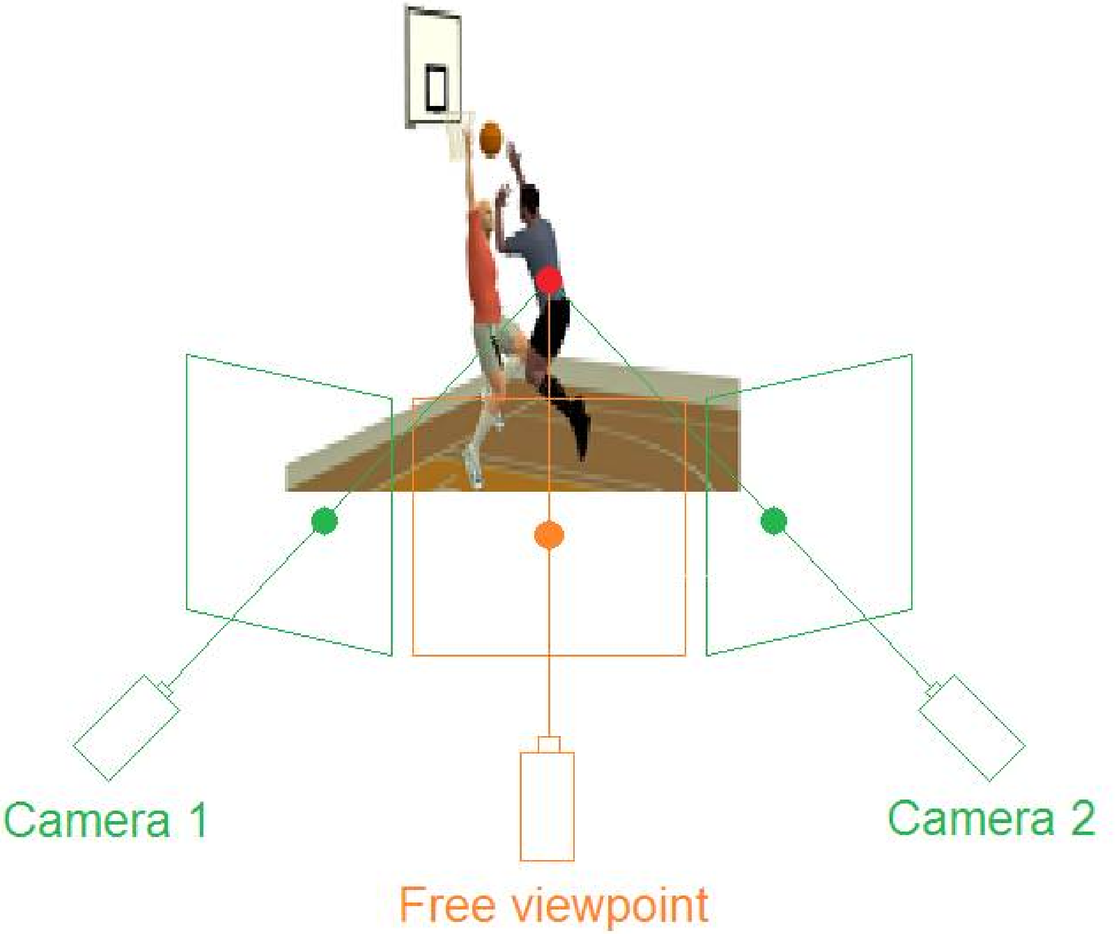
Conventional video production fixes the viewpoint to one of the cameras that capture the scene

Weaknesses of existing solution

Main issues: Restricted camera coverage (cameras on the same side, outside the scene)

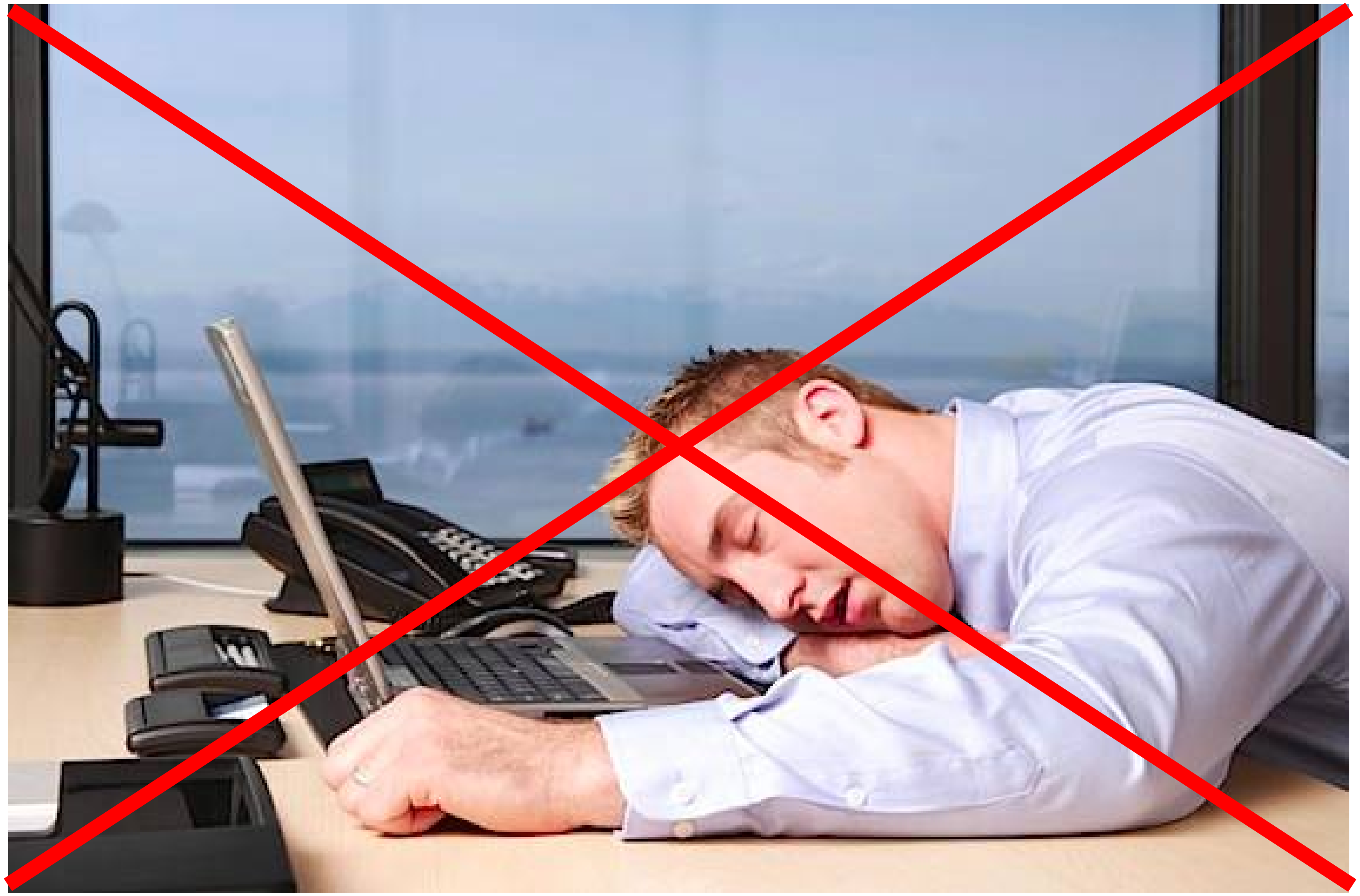
Non-smooth transitions between fixed cameras.

What is free viewpoint rendering ?



The concept of FVR is born with the Matrix movie

It works perfectly, so what do you do at work ?



The making-of : from Matrix to matrix...

- 120 precisely mounted and synchronized cameras
- Hundreds of man-hours to make the transition smoother [1]

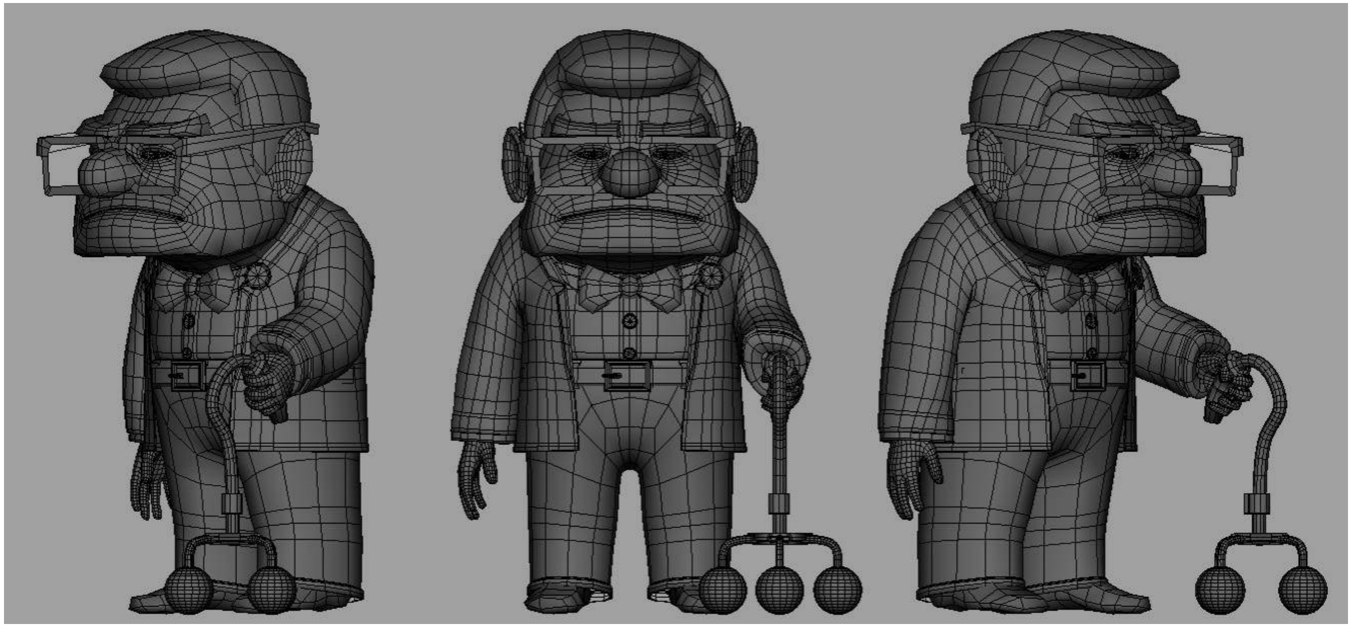
➡ How to automatically render a virtual view ?

Organization of the presentation

- Context
- Acquisition setups
- Rendering virtual views from images
- Rendering virtual views from 3D model
- Image-based rendering vs model-based rendering

The 3D scene must be known to render a virtual view

- A virtual view is a projection of the 3D scene onto an image



What if the 3D of the scene is not fixed ?

The 3D of the scene can be MEASURED (1)

Infrared camera

...

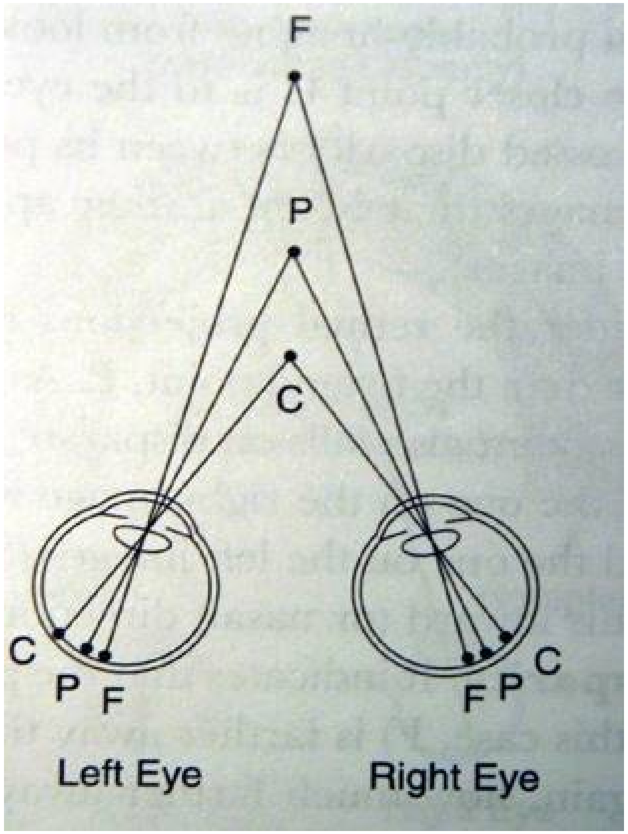
Specific time-of-flight camera

$

Depth range

➡ Limited depth range

The 3D of the scene can be MEASURED (2)



P : converging point

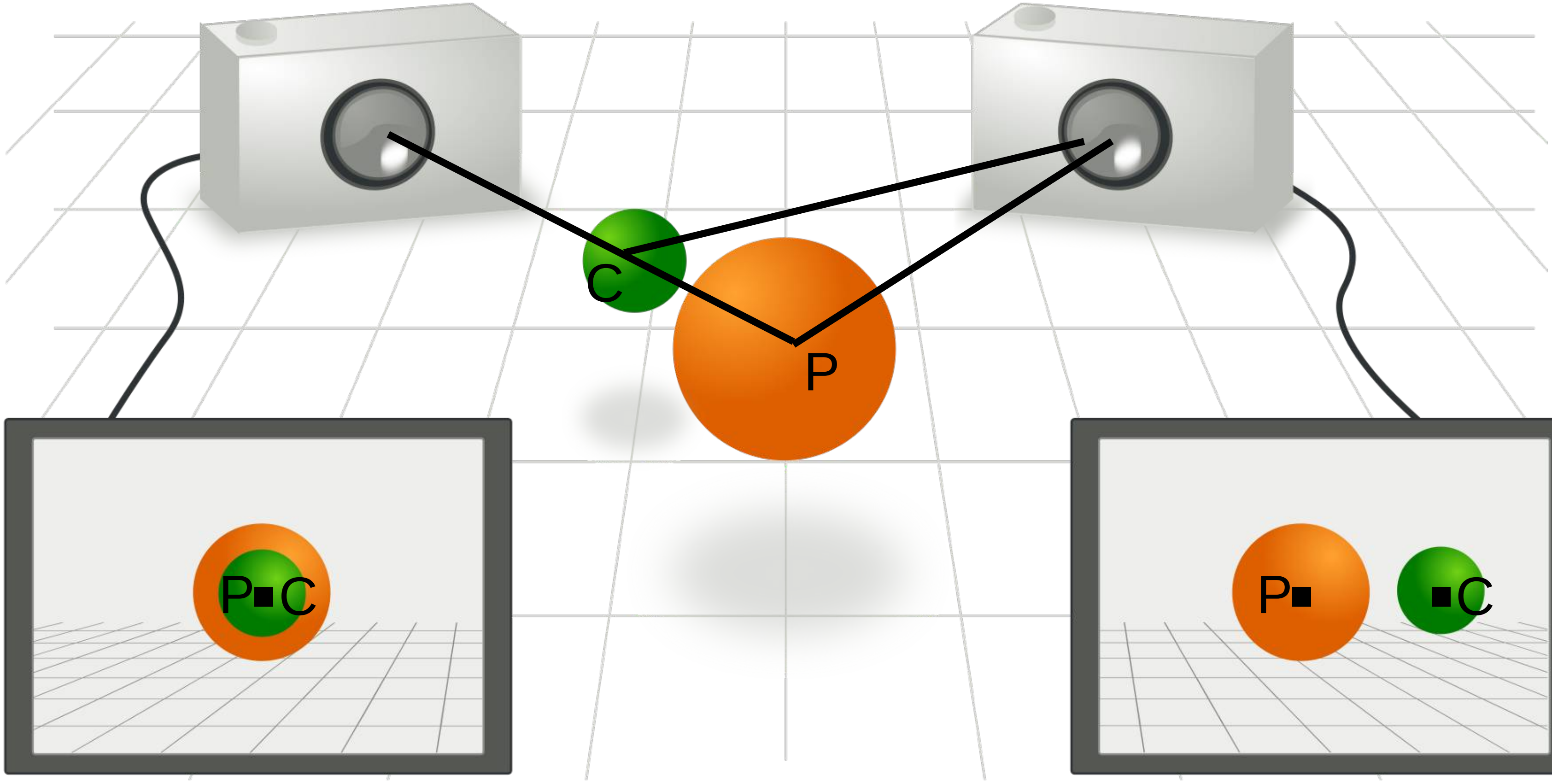
C : object **C**loser projects to the outside of P

F : object **F**urther projects to the inside of P

The 3D of the scene can be COMPUTED by mimicing the human vision

P : converging point

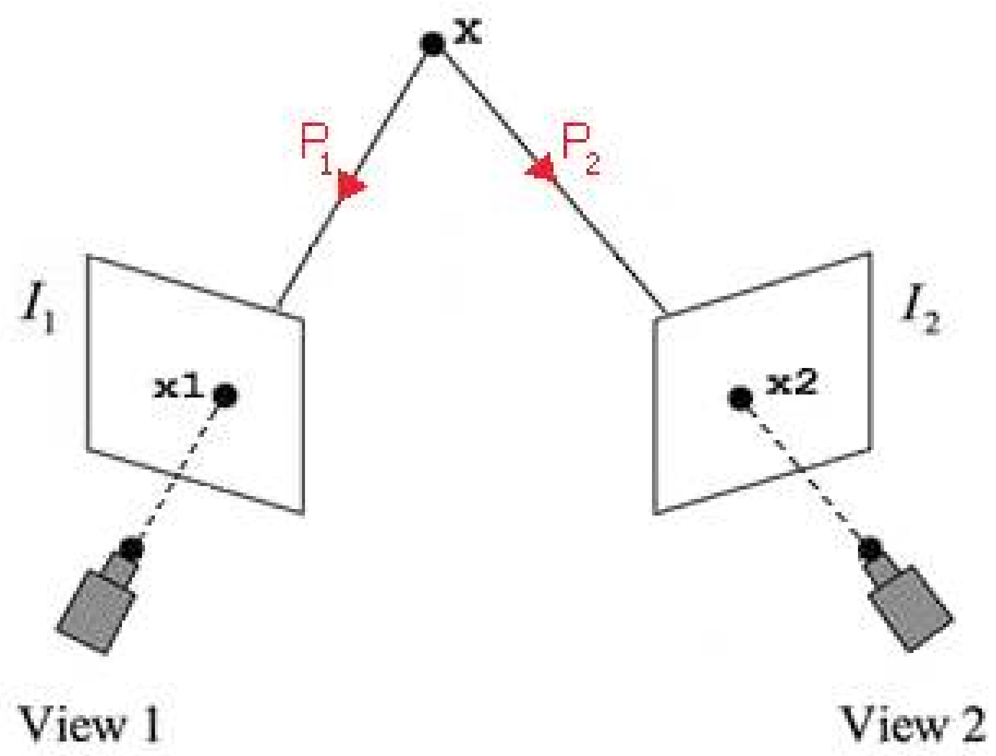
C : object **C**loser projects to the outside of P



Organization of the presentation

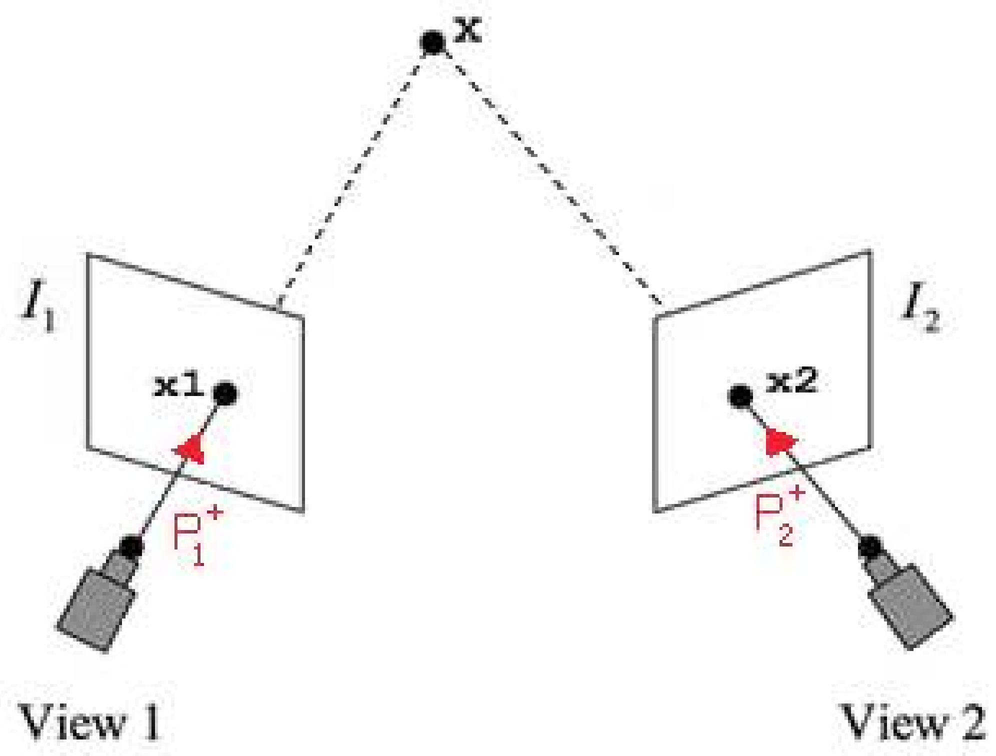
- Context
- Acquisition setups
- Rendering virtual views from images
- Rendering virtual views from 3D model
- Image-based rendering vs model-based rendering

A (pin-hole) camera projects the rays of light on the image plane



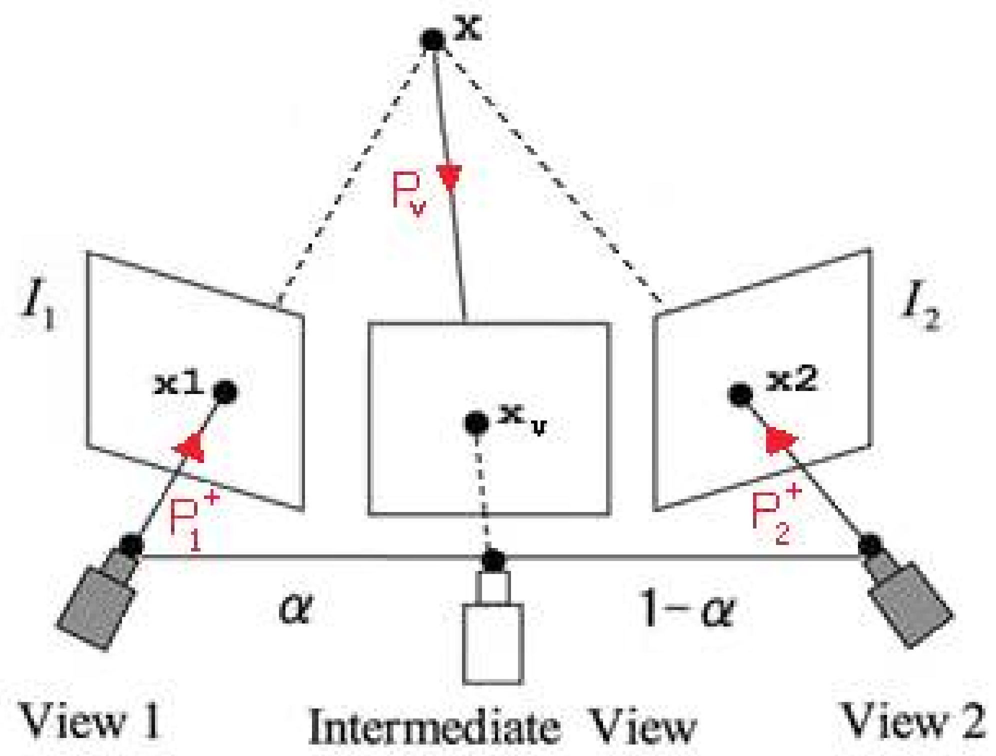
The projection matrices (P1 and P2) are determined by camera calibration [2]

By determinating pixel correspondences, the 3D scene can be reconstructed by triangulation



$P1^+$ and $P2^+$ are the pseudo-inverses of P1 and P2 [3]

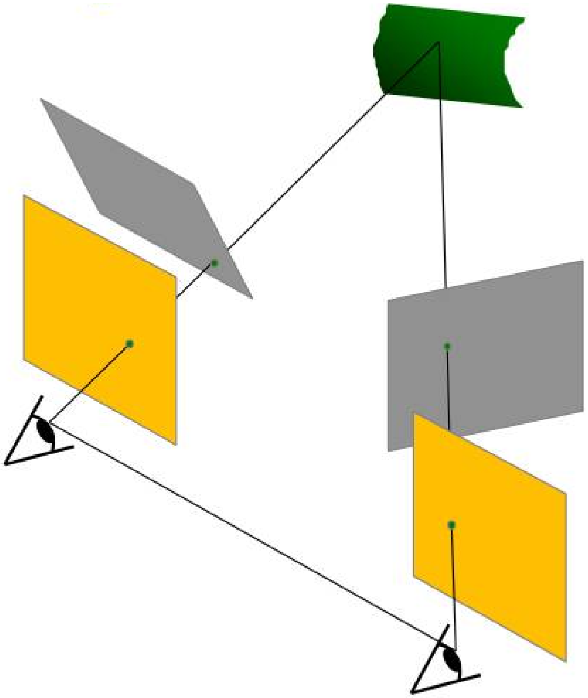
This reconstructed 3D scene can be projected on the virtual view

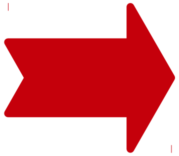


$P_V = K_V[R_V \mid T_V]$ is obtained by “linear interpolation” of the parameters of the reference cameras [4]

Finding pixel correspondences is a computationally expensive process

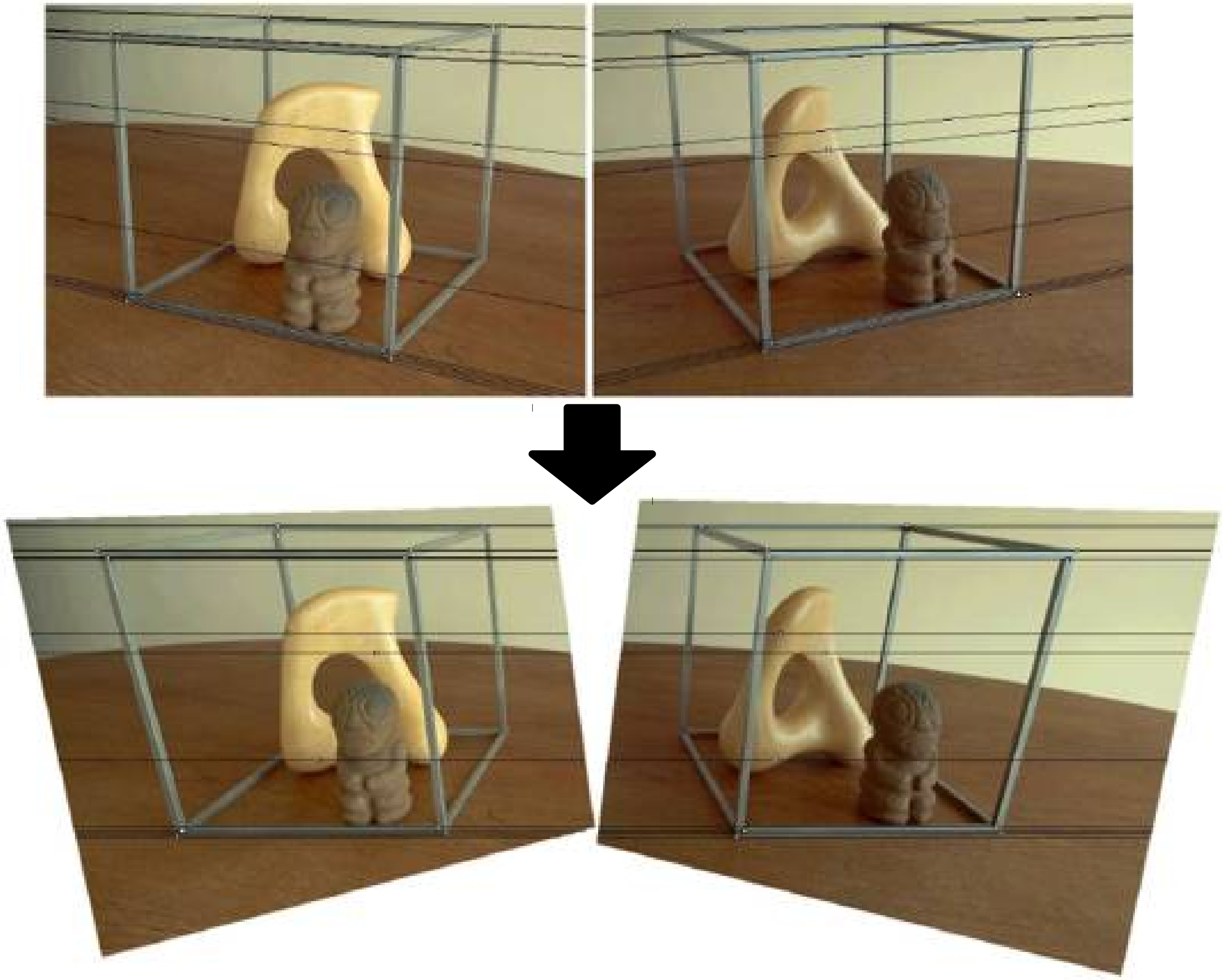
- Reproject the image planes onto the common plane parallel to the line between the optical centers [5]





Reduction to a 1D matching problem

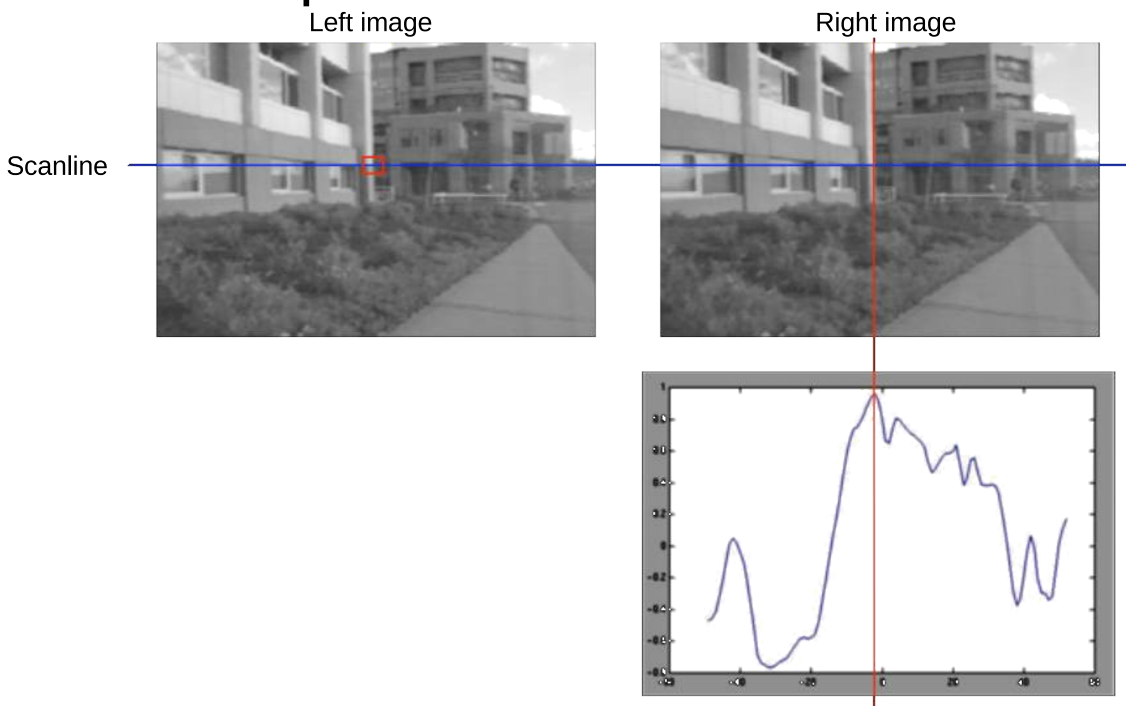
An example of epipolar lines rectification



A good tool to create panoramas

Find pixel correspondences : last step, but not least !

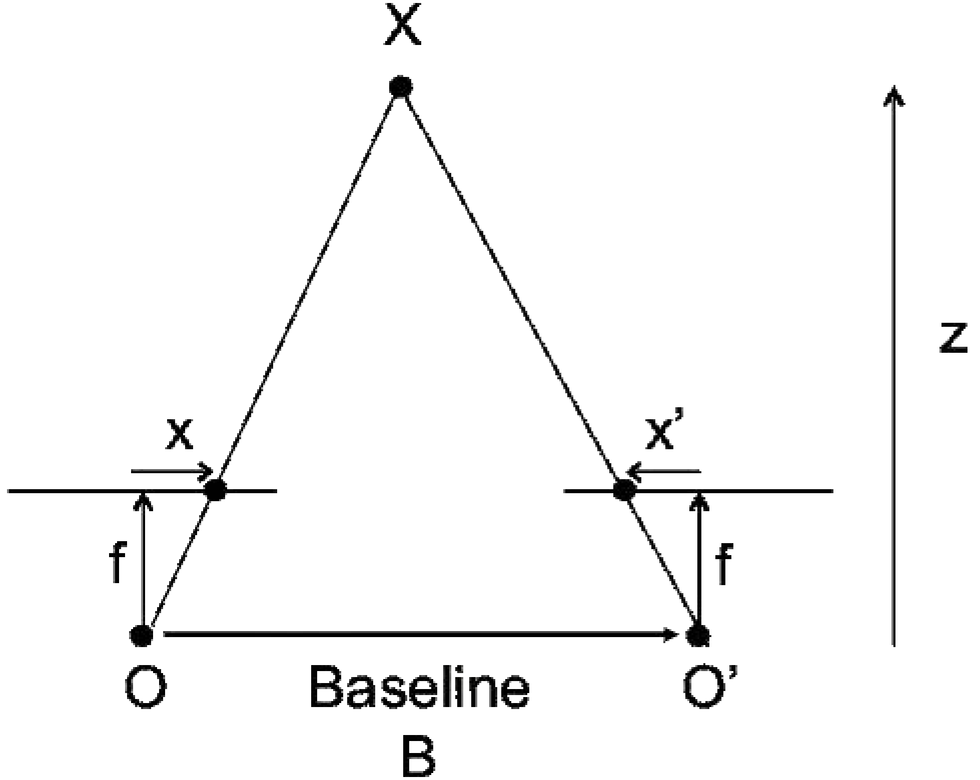
• Local methods (block matching : SAD, NCC, census [6], AD-census [7], etc.) : slide a window along the right scanline and compare contents with the reference window



• Non-local methods (graph-cuts [8], dynamic programming [9], belief-propagation [10], etc.) : based on uniquess, ordering and smoothness

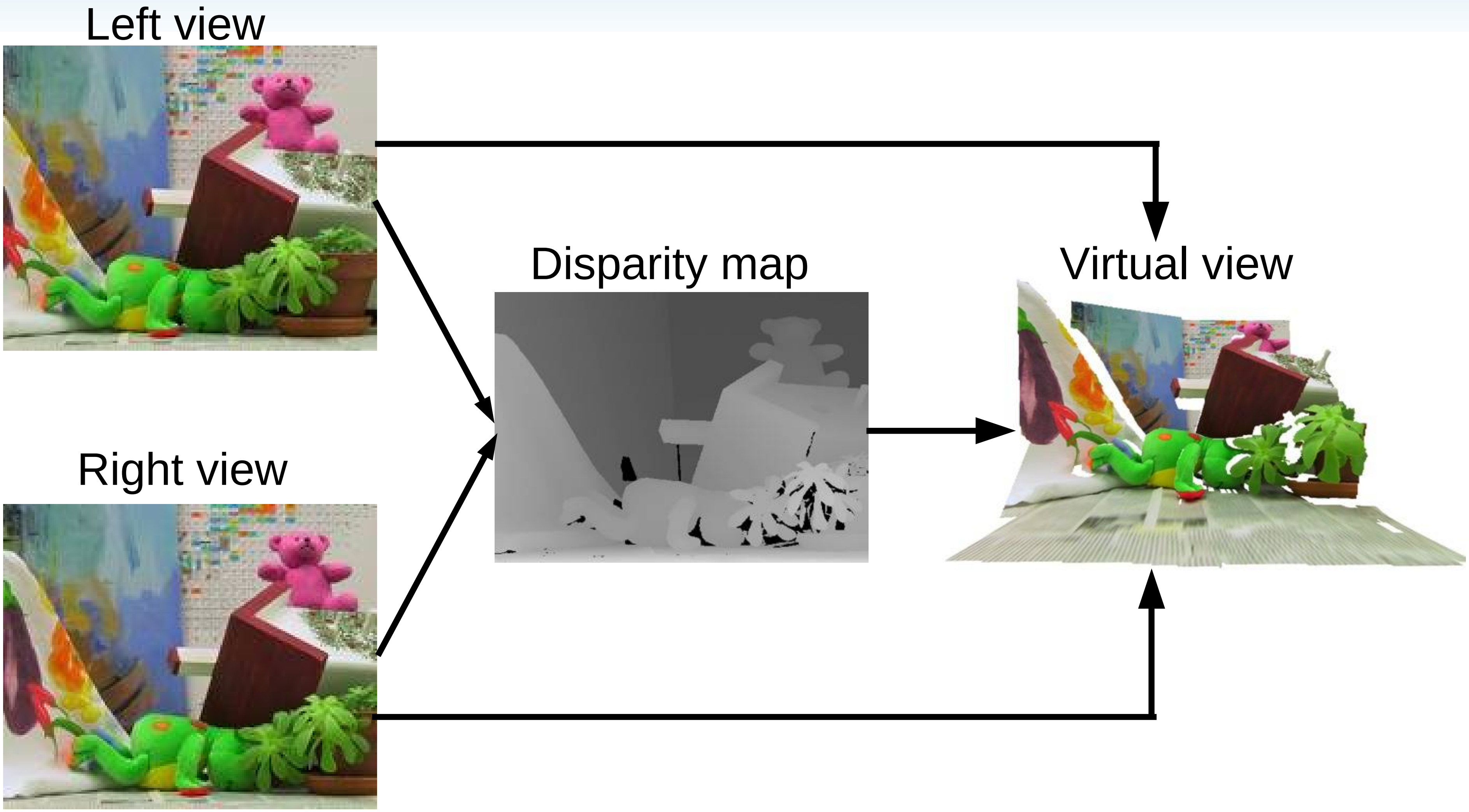
Computing the depth from the disparity

- The difference in position between the corresponding pixels in two images is called disparity



$$disparity = x - x' = \frac{B \times f}{z}$$

In summary



Take part in the competition (more than 140 competitors from now) [11]!

Error Threshold = 1		Sort by nonocc				Sort by all				Sort by disc				
Algorithm	Avg. Rank	Tsukuba ground truth			Venus ground truth			Teddy ground truth			Cones ground truth			Average percent of bad pixels (explanation)
		nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	
AdaptGCP [137]	6.8	1.03 13	1.29 5	5.60 15	0.10 3	**0.14** 1	1.30 4	4.63 15	6.47 6	12.5 17	**1.81** 1	**5.70** 1	**5.33** 1	3.83
ADCensus [94]	9.9	1.07 17	1.48 14	5.73 20	0.09 2	0.25 8	1.15 3	4.10 10	6.22 4	10.9 9	2.42 11	7.25 9	6.95 12	3.97
AdaptingBP [17]	12.3	1.11 21	1.37 8	5.79 22	0.10 4	0.21 5	1.44 6	4.22 12	7.06 9	11.8 13	2.48 13	7.92 17	7.32 17	4.23
CoopRegion [41]	12.4	0.87 4	**1.16** 1	4.61 4	0.11 5	0.21 4	1.54 8	5.16 23	8.31 14	13.0 20	2.79 26	7.18 8	8.01 32	4.41
RVbased [116]	16.0	0.95 9	1.42 12	4.98 9	0.11 7	0.29 13	**1.07** 1	5.98 31	11.6 42	15.4 38	2.35 9	7.61 10	6.81 11	4.88
DoubleBP [35]	16.5	0.88 6	1.29 4	4.76 7	0.13 9	0.45 31	1.87 15	3.53 7	8.30 13	9.63 6	2.90 32	8.78 42	7.79 26	4.19
RDP [102]	16.6	0.97 10	1.39 10	5.00 10	0.21 28	0.38 21	1.89 16	4.84 16	9.94 25	12.6 18	2.53 15	7.69 12	7.38 18	4.57
OutlierConf [42]	17.6	0.88 5	1.43 13	4.74 6	0.18 19	0.26 11	2.40 28	5.01 19	9.12 20	12.8 19	2.78 25	8.57 33	6.99 13	4.60
SurfaceStereo [79]	22.5	1.28 35	1.65 23	6.78 42	0.19 21	0.28 12	2.61 40	3.12 4	**5.10** 1	8.65 2	2.89 31	7.95 19	8.26 40	4.06
SubPixDoubleBP [30]	22.8	1.24 29	1.76 34	5.98 26	0.12 8	0.46 33	1.74 12	3.45 6	8.38 15	10.0 8	2.93 35	8.73 39	7.91 28	4.39
SubPixSearch [127]	22.9	2.04 79	2.48 68	6.40 35	0.14 13	0.40 25	1.74 12	4.00 9	6.39 5	11.0 11	2.24 6	6.87 6	6.50 6	4.18
LLR [135]	24.2	1.05 14	1.65 22	5.64 16	0.29 47	0.81 62	3.07 49	4.56 14	9.81 24	12.2 14	2.17 3	8.02 21	6.42 4	4.64
WarpMat [55]	26.3	1.16 22	1.35 7	6.04 27	0.18 20	0.24 7	2.44 32	5.02 20	9.30 21	13.0 22	3.49 51	8.47 31	9.01 56	4.98
ObjectStereo [98]	27.6	1.22 28	1.62 18	6.36 33	0.59 75	0.69 54	4.61 76	4.13 11	7.59 10	11.2 12	2.20 4	6.99 7	6.36 3	4.46
PMF [138]	29.9	1.74 63	2.04 53	8.07 71	0.33 53	0.49 36	4.16 69	**2.52** 1	5.87 3	**8.30** 1	2.13 2	6.80 5	6.32 2	4.06
HEBF [123]	30.9	1.10 20	1.38 9	5.74 21	0.22 29	0.33 18	2.41 30	6.54 51	11.8 46	15.2 35	2.78 24	9.28 54	8.10 34	5.41
PatchMatch [112]	31.1	2.09 81	2.33 64	9.31 80	0.21 26	0.39 23	2.62 41	2.99 3	8.16 11	9.62 5	2.47 12	7.80 13	7.11 14	4.59
GC+SegmBorder [57]	32.4	1.47 51	1.82 36	7.86 65	0.19 22	0.31 14	2.44 32	4.25 13	5.55 2	10.9 10	4.99 93	5.78 2	8.66 49	4.52
PMBP [131]	32.7	1.96 73	2.21 61	9.22 78	0.30 48	0.49 34	3.57 63	2.88 2	8.57 16	8.99 3	2.22 5	6.64 4	6.48 5	4.46
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

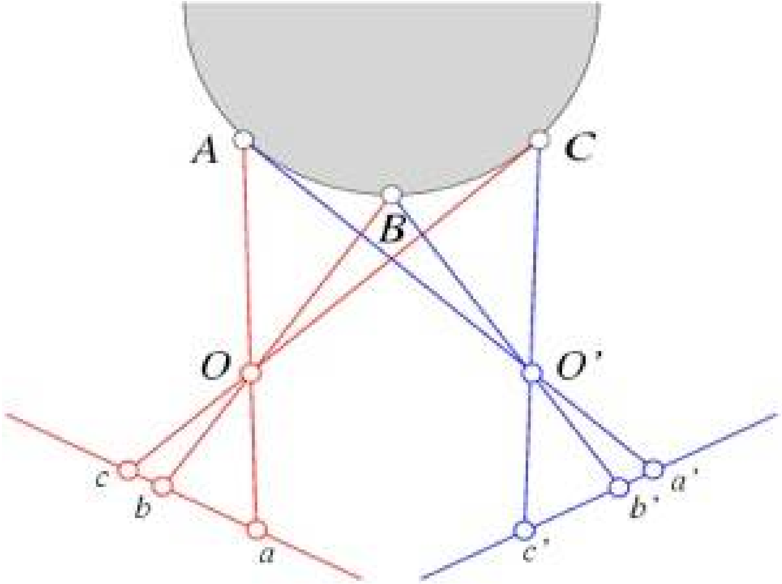
Organization of the presentation

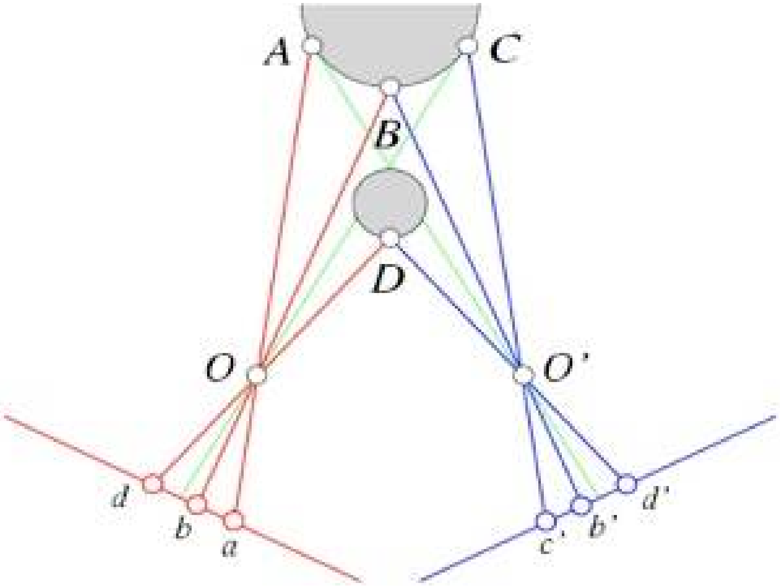
- Context
- Acquisition setups
- Rendering virtual views from images
- Rendering virtual views from 3D model
- Image-based rendering vs model-based rendering

Virtual view synthesis is not as simple when the cameras are far away from each others (1)

- <u>Large occlusions</u> make the part-to-part correspondences impossible

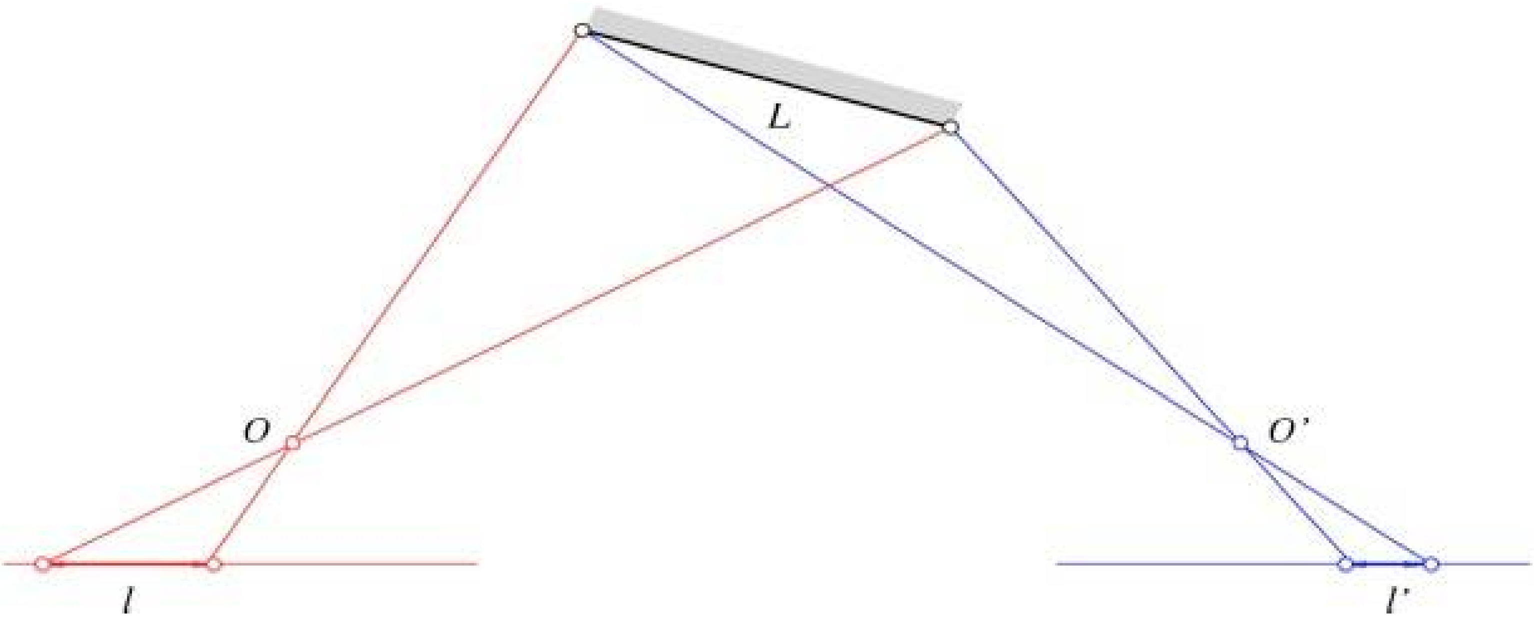
Virtual view synthesis is not as simple when the cameras are far away from each others (2)



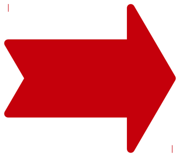


- The <u>ordering constraint</u> (used in non-local methods) does not hold anymore

Virtual view synthesis is not as simple when the cameras are far away from each others (3)



- The matching with fixed-size windows fails because of <u>foreshortening</u>

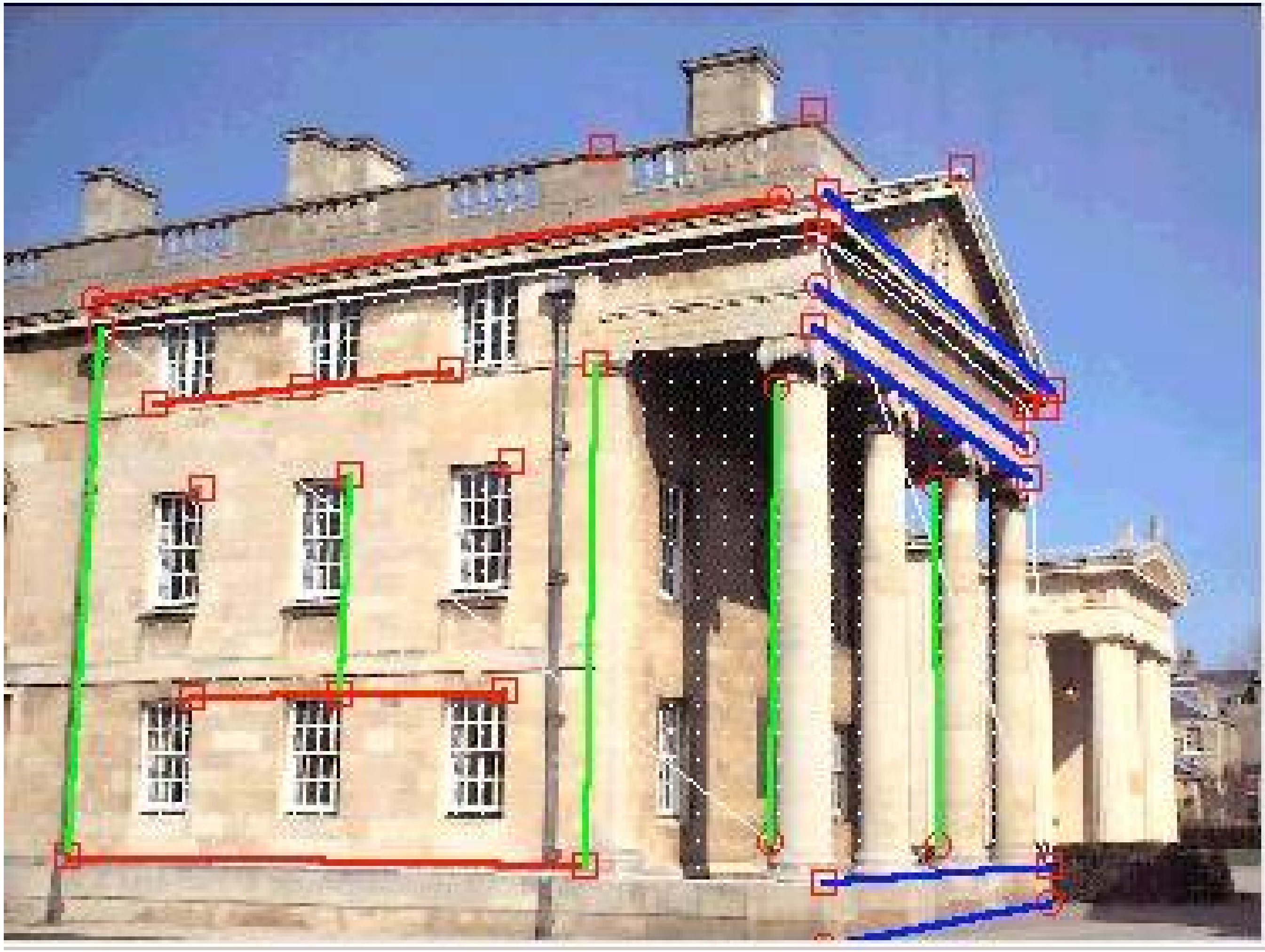


Reconstruct explicitly the 3D model of the scene

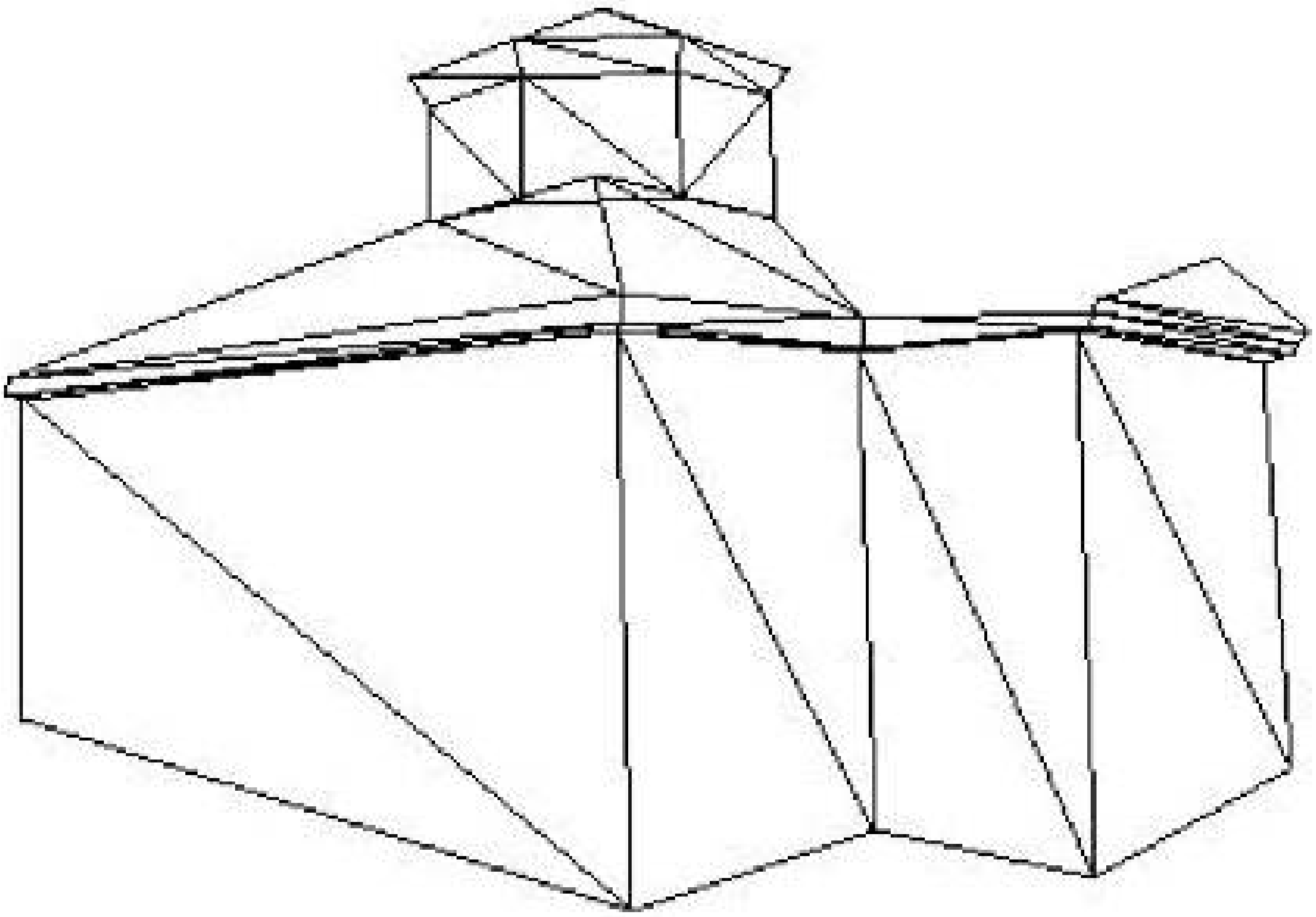
=

Model-based rendering

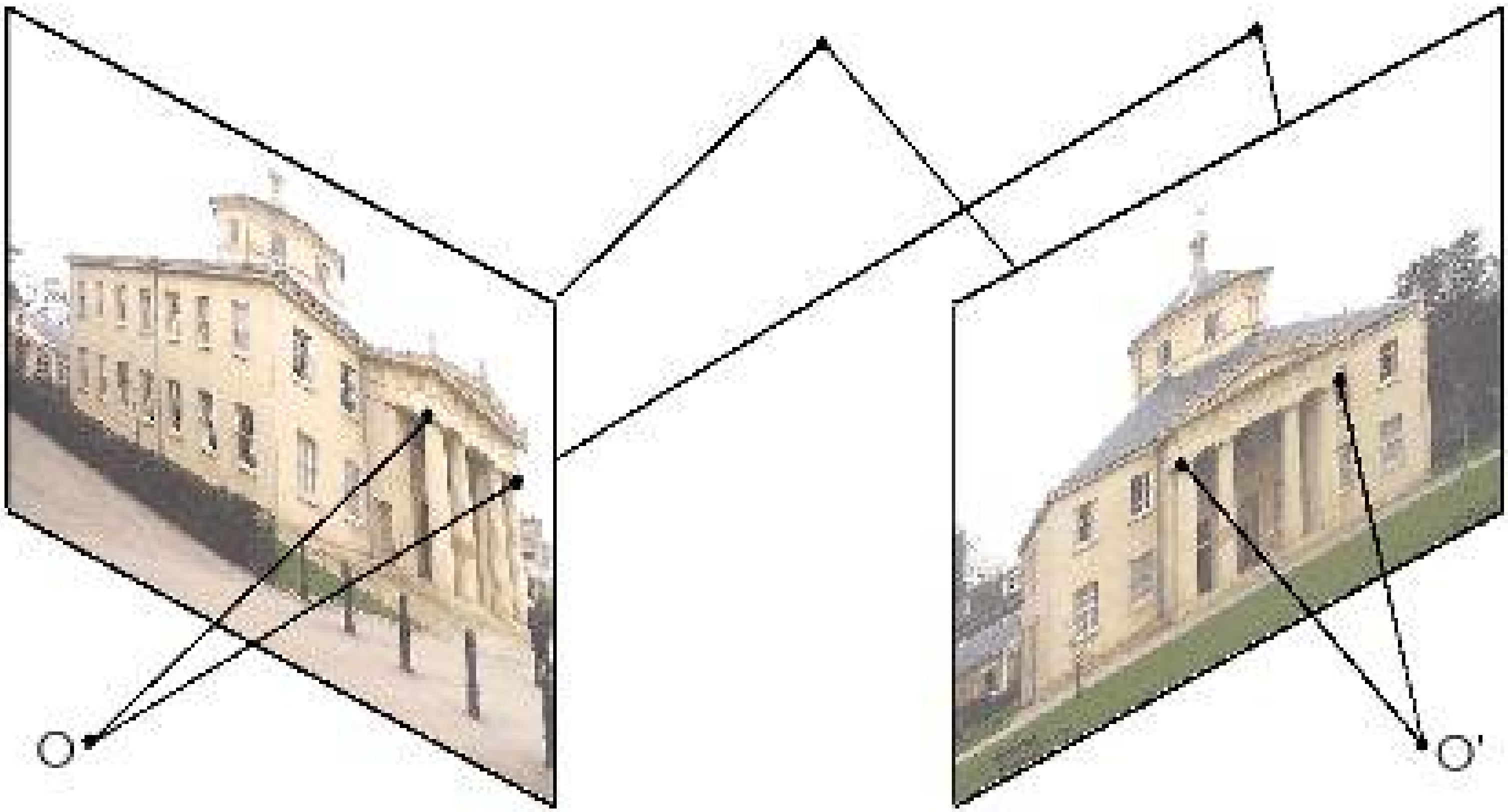
Manually select corresponding features...



...and define a polygonal model [3]

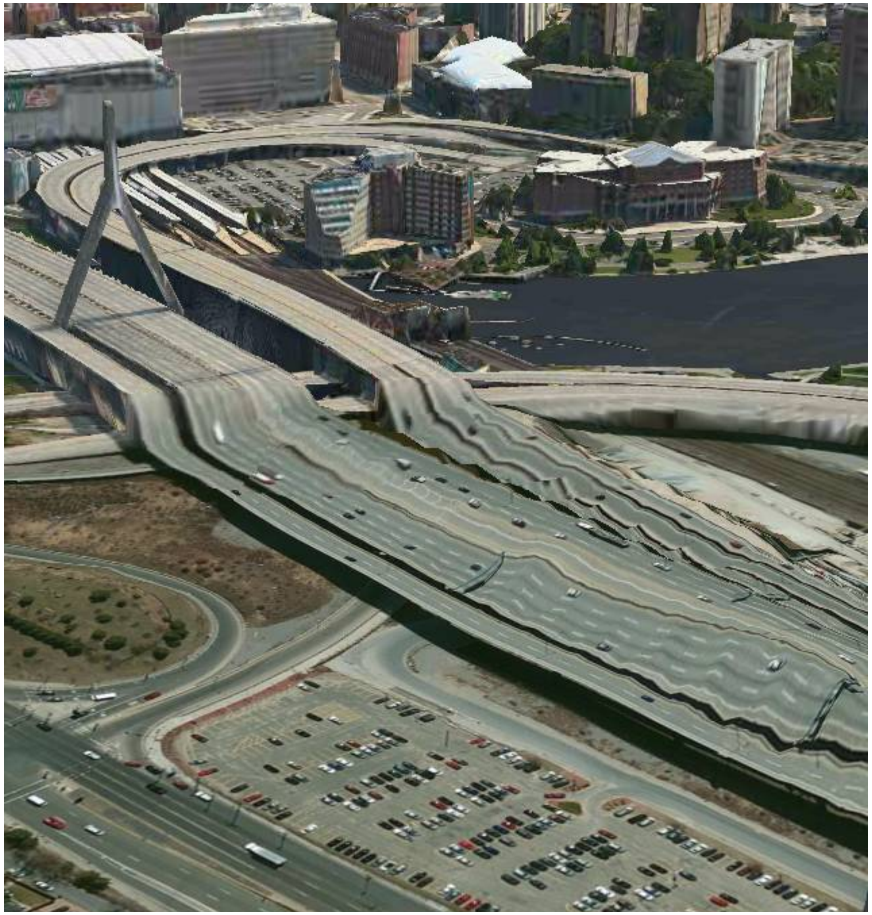


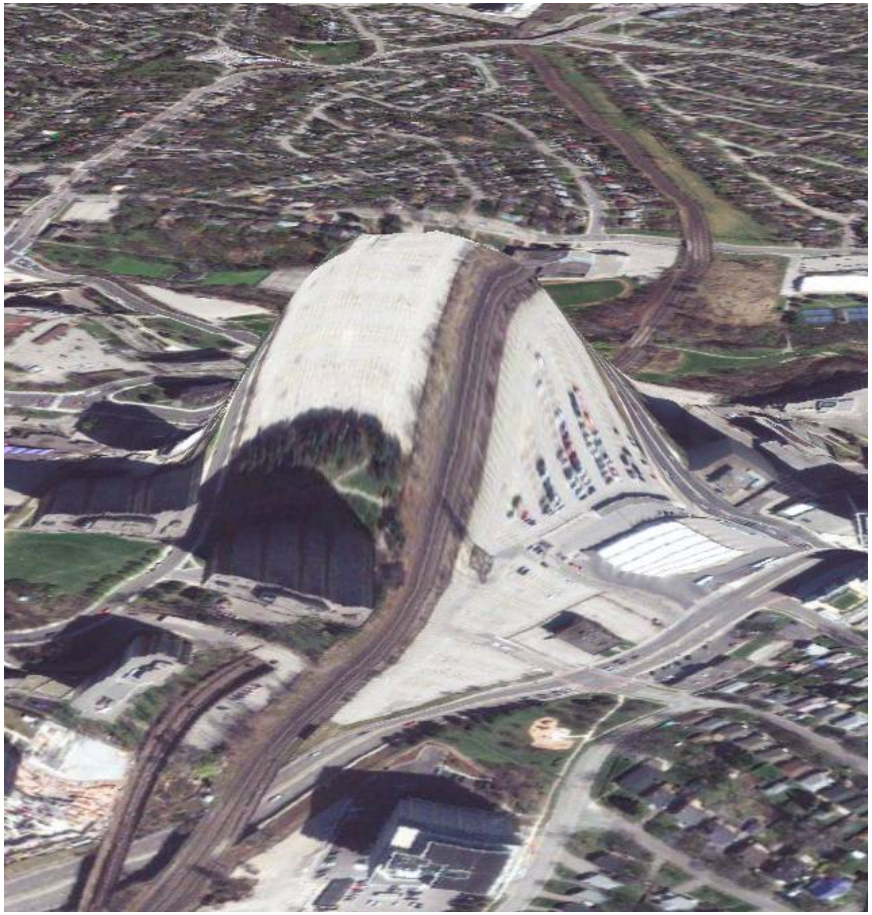
Correct the 3D model thanks to the epipolar geometry (*bundle adjustment [12]*)



And finally add the textures (*texture wrapping*)

Cross your fingers for a good model (1)





Cross your fingers for a good model (2)

Organization of the presentation

- Context
- Acquisition setups
- Rendering virtual views from images
- Rendering virtual views from 3D model
- Image-based rendering vs model-based rendering

Comparison between image-based rendering and model-based rendering

Image-based rendering	Model-based rendering
• Slow (pixel matching)	• Fast (precomputed 3D model)
• Not suitable for large baseline	• Suitable for large baseline
• Automatic (variable 3D model)	• Manual (fixed 3D model)

➡ Any rivalry between image-based proponents and model-based proponents is non-sense, because they are COMPLEMENTARY methods

A good example

Conclusion

- Free viewpoint rendering enables to virtually navigate across a scene
- The 3D of the scene must be reconstructed, in such a way to project it onto the virtual view
- Two types of methods are used for 3D reconstruction : image-based rendering and model-based rendering.
- The constant war between their proponents is non-sense, because they are complementary methods
- Perspective : regularization of image-based rendering with an (learnt) *a priori* of the projection of the 3D model

Some references

(1) N. Inamoto and H. Saito, "Virtual viewpoint replay for a soccer match by view interpolation from multiple cameras" in *IEEE Trans. on Multimedia (TM07)*, vol.9(6), p.1155-1166, 2007.

(2) Z. Zhang, "A flexible new technique for camera calibration" in *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI'00)*, vol 22(11), p.1330-1334, 2000.

(3) R. Hartley and A. Zisserman, ''Multiple View Geometry in Computer Vision'', Cambridge University Press, Cambridge, UK, 2000.

(4) J. Park, "Quaternion-based camera calibration and 3D scene reconstruction", in *Fourth ACM/IEEE International Conference on Computer Graphics, Imaging and Visualization (IJCV07), p. 89-92, 2007.*

(5) C. Loop and Z. Zhang, "Computing Rectifying Homographies for Stereo Vision", *in International Conference on Pattern Recognition (CVPR '99),* vol 1, 1999.

(6) H. Hirschmüller and D. Scharstei*n, "Evaluation of stereo matching costs on images with radiometric differences", in IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI'09),* vol 30(2), p.1328-341, 2009.

(7) X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang and X. Zhang, "On building an accurate stereo matching system on graphics hardware", *in IEEE International Conference on Computer Vision Workshops (ICCVW'11),* p.467-474, 2011.

(8) Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization via Graph Cuts", in *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI'01),* vol 23(11), p.1222-1239, 2001.

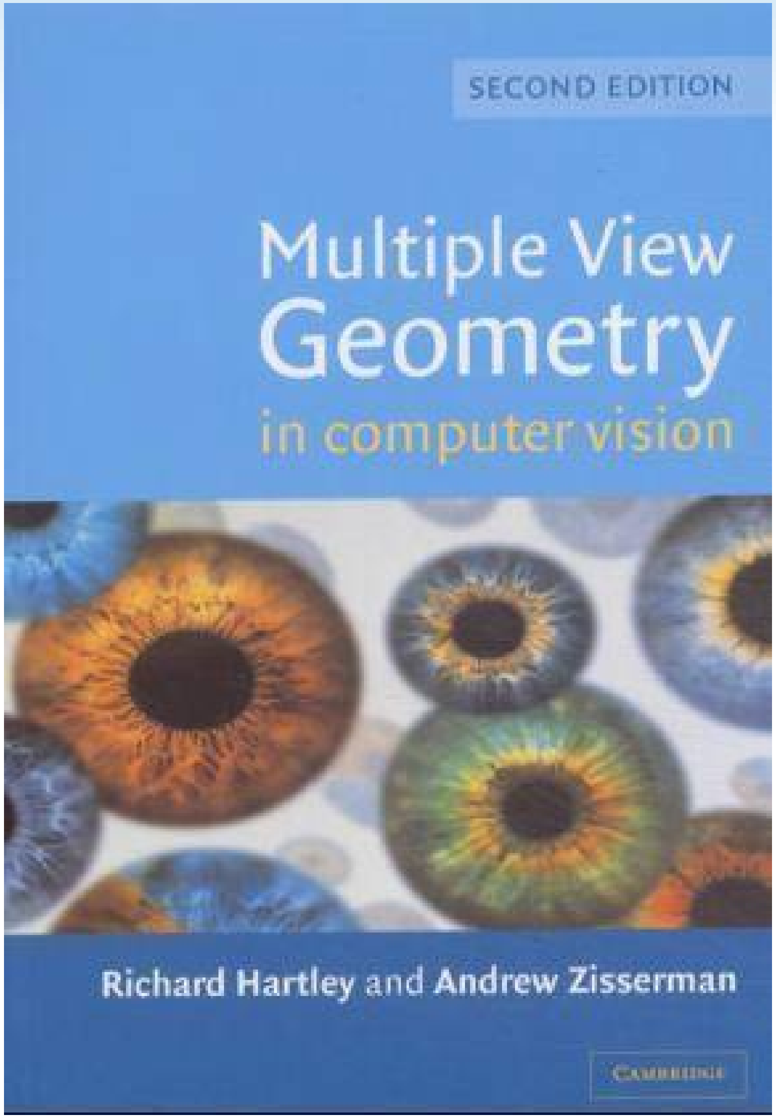
(9) I.J. Cox and S.L. Hingorani and S.B. Rao and B.M. Maggs, ''A maximum likelihood stereo algorithm'', *in Computer Vision and Image Understanding (CVIU'96),* vol.96, p.542-567, 1996.

(10) A. Klaus, M. Sormann and K. Karner, "Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure", in 18th Int. Conf. on. Pattern Recognition (ICPR'06), vol 3, p.15-18, 2006.

(11) D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms", *in International Journal of Computer Vision (IJCV'02)*, vol. 47(1:3), p.7-42, 2002.

(12) B. Triggs, P. McLauchlan, R. Hartley and A. Fitzgibbon, "Bundle adjustment – a modern synthesis", Springer, in Vision algorithms: theory and practice, p.153-177, 2000.

Some references



Thank you very much for your attention...

Questions?