Feature Point based object tracking

AMI training program report

Jozef Mlích
imlich@fit.vutbr.cz
DCGM, FIT, BUT

Professor Dr.-Ing. habil. Gerhard Rigoll, rigoll@tum.de
Dipl.-Ing. Benedikt Hörnler, b@tum.de

2009

The video content understanding is very important for various applications, e.g. meeting analyses. State of the art approaches often try to spot object directly in videos, track them in time and detect the events in its trajectory. This work focuses on object detection and tracking on corner points. More accurately, the proposed approach deals with point of interest based background modelling and foreground segmentation. The foreground objects should be modelled as graph of salient points, which should accurately represent the object. Each node should be denoted by its position and descriptor based on additional feature, as gradients, colour histogram, etc, while the graph's edges are represented by node distances and directions. The detected object graph can subsequently be tracked by applying graph similarity measures. More robustness can be gained by adding some elasticity into the graph structure, which is important for non-rigid objects. Finally, the algorithm should be extended to the 3D space and applied to several synchronised cameras.

1 Introduction

Object recognition and tracking is central problem of computer vision. A lot of algorithms for scene understanding and knowledge mining related to surveillance or meeting data analysis requires temporal information about object position [11, 10]. Common approaches are based on localisation of previously defined object[7] and its tracking in time. More general algorithms should track previously undefined objects. For such tasks is object usually described by pixels in picture and its position extracted according to background model [4].

Description of object by all of its pixels is indeed not very practical regarding to spatial properties of real world objects and amount of necessary data. One of solutions for this problems is describing object by some salient points[3]. Afterwards, for recognition of independent objects in point clouds the clustering and segmentation techniques are used. This problem is discussed in section 2.

The main problem of classical tracking algorithms based on foreground models is object occlusion. The solution of occlusion problem could be based on description of object or on adding of additional information from camera with different field of view. The multiple camera point localisation is discussed in section 3.

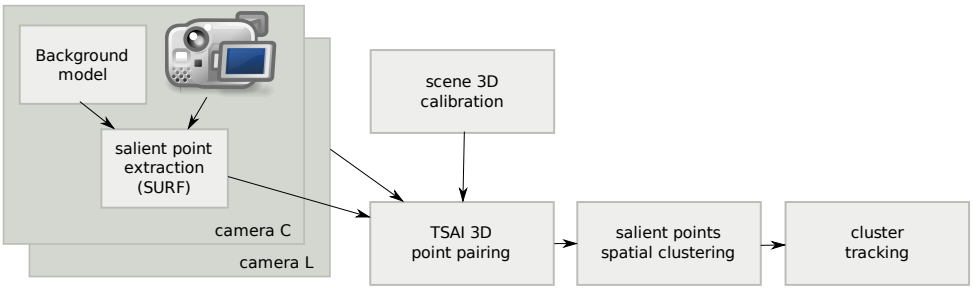

Figure 1: Proccesing pipeline

The processing pipeline illustrating proposed approach is shown in Figure 1.

2 Salient points

The objects could be effective described by a few points. For purpose of object recognition is point defined by its position and by descriptor. Position of points could be selected by corner point detector, e.g. by Harris corner detector. Descriptor is feature vector with describes appearance of small neighbourhood of point. The descriptor could contain a couple of different features, e.g. colour histogram, local binary pattern, local rank differences, gradient histogram, etc. The requirements on salient points extraction algorithm are especially algorithm speed, temporal stability, scalability, rotation invariance and illumination invariance.

The source image should indeed also met some minimal requirements. To ensure temporal stability is recommended to process deinterlaced video sequence or video sequence recorded with progressive scan.

The salient point localisation and feature vector extraction is expansive operation. However, there exists a lot of algorithms suitable for object localisation and object tracking [5]. The OpenCV [8] implementation of SURF algorithm offers fast point localisation and heterogeneous feature vector, additionally it is easy replaceable by OpenCV MSER[9] implementation.

Detailed description over SURF feature points is in [3]. For point detec-

tion is used fast Hessian algorithm, which is made over integral image. The descriptor is made by first order Haar responses in directions x and y.

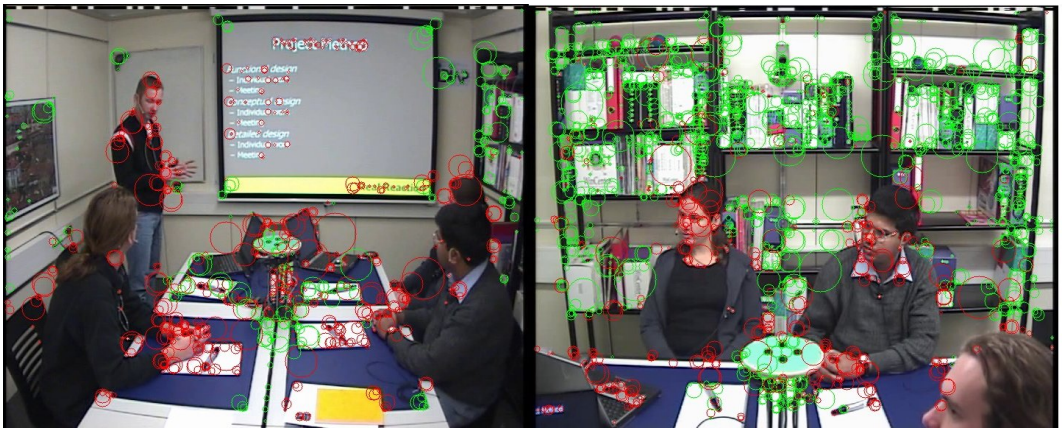

Figure 2: IDIAP SMART meeting room scene setup with foreground points

Standard background models are usually build over all pixels in image. The models based on feature points [15] are very similar. In image is only a couple of most significant feature points. The background model could contain descriptors for all points in image or could be sparse. In case of sparse model the assumption is, that the amount of a couple most significant points is sufficient for background model, e.g. match of observed point with background point means automatically that point belongs to background.

The result of foreground modelling based on SURF points is shown in Figure 2. The green circles denotes background points and red circles denotes foreground points. The size of circle denotes scale parameter of SURF points.

3 Camera calibration

The goal is to find corresponding objects in overlapped parts of views of cameras i.e. fundamental matrix of views in epipolar geometry [12]. It is necessary to calibrate the scene by defining image coordinates[1] and corresponding real world coordinates[2].

[1] 2D coordinates in image
[2] 3D coordinates

However, epipolar geometry could be applied only in case of undistorted images. In stereo vision, the lens distortion are defined as intrinsic camera parameters. The fundamental matrix of epipolar geometry is defined as extrinsic parameters.

It is possible to calibrate scene in semi automatic way [14, 8], however this is possible only with recordings with calibration chessboard. The Tsai algorithm[13] with proper calibration data allows to find intrinsic and extrinsic parameters for scene. The calibration data was prepared by hand according to Schematics[1], AMIDA Deliverable D2.2 and additional by hand made measurements obtained directly from IDIAP.

The calibration data for scene are attached in table 3. The calibration was done on recordings labled IS1000a-C.avi, IS1000a-L.avi, see in Figure 1. Lens distortion compensation is shown in Figure 3.

Figure 3: Lens distorsion compenstaion applied on image obtained from camera C (left) and camera L (right)

In case of planar objects it is possible to use homography for computing of points correspondence. However, real world object are non planar. It is possible to divide scene in multiple layers and search homography separately for each object [2].

4 Object tracking

Definition of object itself in general is hard. In some cases (i.e. scenes and situations) is possible to define object by template or model (i.e. generalised template). In contrary to that, we try to spot unknown object in scene, build its model and track object in time.

Automatic construction of unknown object model is difficult problem. 3D objects are after scanning by camcorder described as pixels in 2D image, therefore is the information about object incomplete. This problem could be solved by scanning with multiple cameras. Object could be partially visible also in case of entering or leaving the scene or in case of occlusion by other object. Real world objects are additionally non rigid and changes its appearance in time.

The result of foreground modelling are the salient points. Step from the points to the object is critical for tracking. The tracker should recognise points which belongs to current models, update them and build new models from rest of salient points. The points are considered to be part of object representation if they belong to the same cluster in multiple view. For two cameras we define $[x_C, y_C, x_L, y_L]^N$, whereas the point is defined by x and y coordinates in both cameras and N denotes count of points describing object. For clustering was selected leader algorithm [6], because does not require initial setup of cluster count. In case, the objects are not ocluded, the clustering gives information about object in consequent frame and the tracking is only search of corresponding clusters.

5 Evaluation

For the evaluation it is necessary to annotated the position of the heads during the recorded meetings. The annotation was performed for 1.500 frames, two camera views (camera L and camer C) and 3 person located in the IDIAP smart meeting room.

Next, the Euclidean distance between target (participant) and nearest detected object for camera C only was measured. The distance for all targets is shown in Figures 4, 5 and 6. The Figure contains also a set of adhoc thresholds, which tries to highlight difference between good detection (closer than 20px) and bad detection.

The same measurement was done also with both views. The coordinates

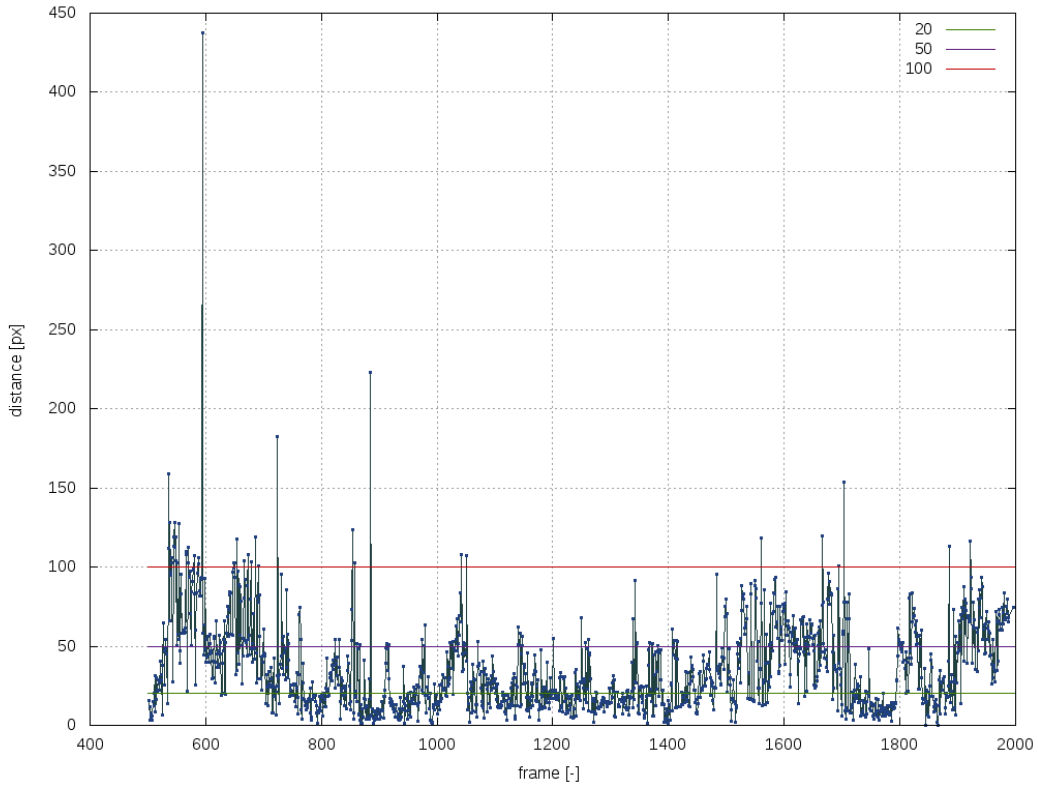

Figure 4: Distance of target 0 from nearest objt in 2D (camera C)

of target are given as two 2D coordinates $[x_C, y_C, x_L, y_L]$ and the the Figures 7, 8, 9 shows Euclidean distance between annotation and nearest object.

The dependency between distance and amount of detection closer than selected threshold is shown in table 1 for 2D detection and table 2 for 2D detection in two views. The data are shown also in Figures 10 and 11.

6 Conclusions

The result of this work includes calibration data for 3D reconstruction in IDIAP smart meeting room based on camera C and camera L. The calibration data are in appendix. Next, the annotation of objects for multiple view tracking evaluation was created. The evaluation of tracking suggests further research of tracking methods. The most challenging task is to find good

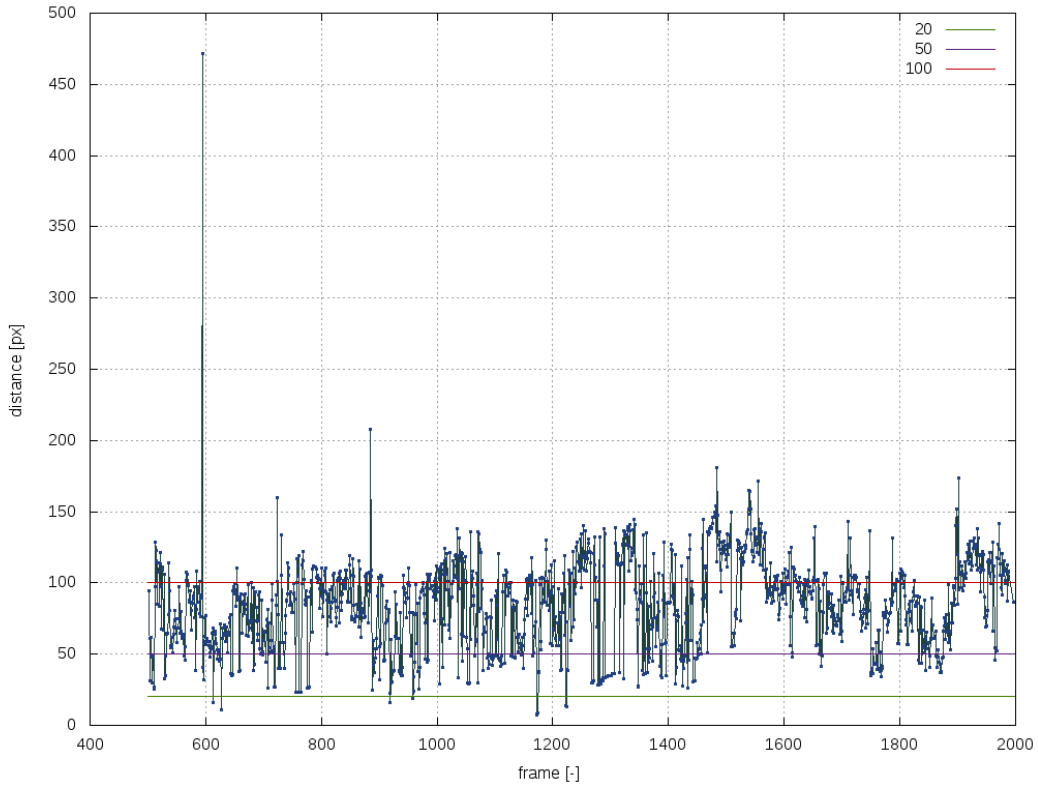

Figure 5: Distance of target 1 from nearest object in 2D (camera C)

feature points for object detection and its recognition in two different views with big angle.

Further research on this topic will be performed during my PhD thesis. The tracking could be markedly improved by including of better spatial comparison of clusters. The tracking system should work better with temporal information. Current system works only with a two consequent frames. Next, processing clusters in two views, this should be extended into n-views tracking system. The salient points are matched in fixed count dimension, therefore objects visible only in single view, are tracked incorrectly. The deployment options are also very limited, because of the manual calibration of multiple view setup and dependency on static background and foreground model. The calibration should be done automatically or semi automatically and the background model should be updated in time. The area for improvements is also in salient point matching problem. The matching of large

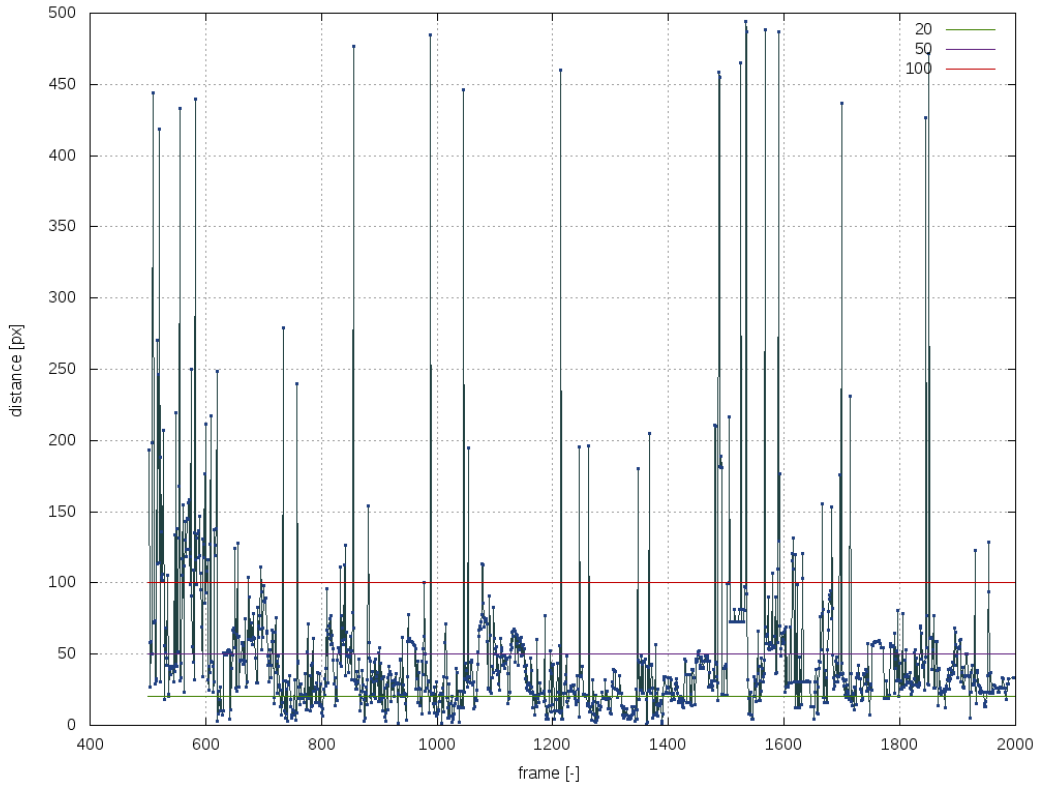

Figure 6: Distance of target 2 from nearest object in 2D (camera C)

set of points is resource consumptive task. Solution for this problem is in simplifying comparison of points or in reduction of amount of points which should be compared i.e. early suppression of object detection.

Acknowledgements

This work has been supported by AMIDA Augmented Multi-party Interaction with Distance Access, EU-6FP-IST, IST-033812-AMIDA.

References

[1] Idiap smart meeting room schematic.

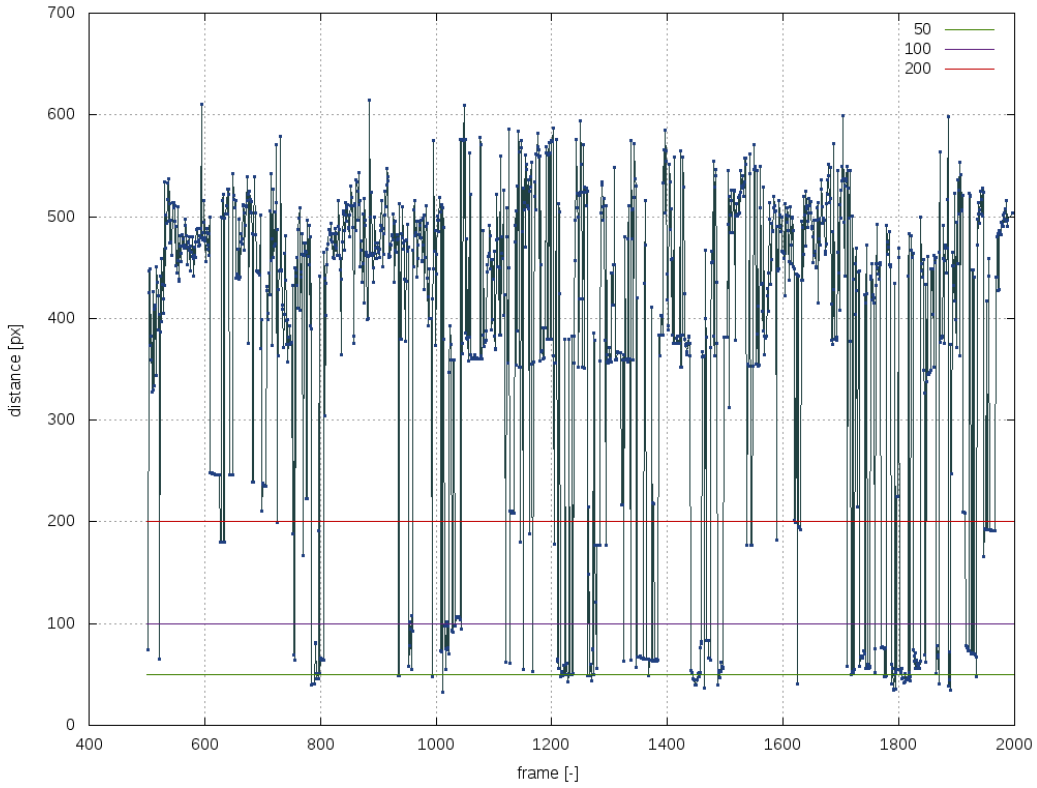

Figure 7: Distance of target 0 from nearest object in two views

[2] D. Arsic, E. Hristov, N. Lehment, B. Hornler, B. Schuller, and G. Rigoll. Applying multi layer homography for multi camera person tracking. pages 1–9, September 2008.

[3] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359, 2008.

[4] Ahmed M. Elgammal, David Harwood, and Larry S. Davis. Non-parametric model for background subtraction. In *ECCV '00: Proceedings of the 6th European Conference on Computer Vision-Part II*, pages 751–767, London, UK, 2000. Springer-Verlag.

[5] A. Haja, S. Abraham, and B. Jahne. A comparison of region detectors for tracking. pages xx–yy, 2008.

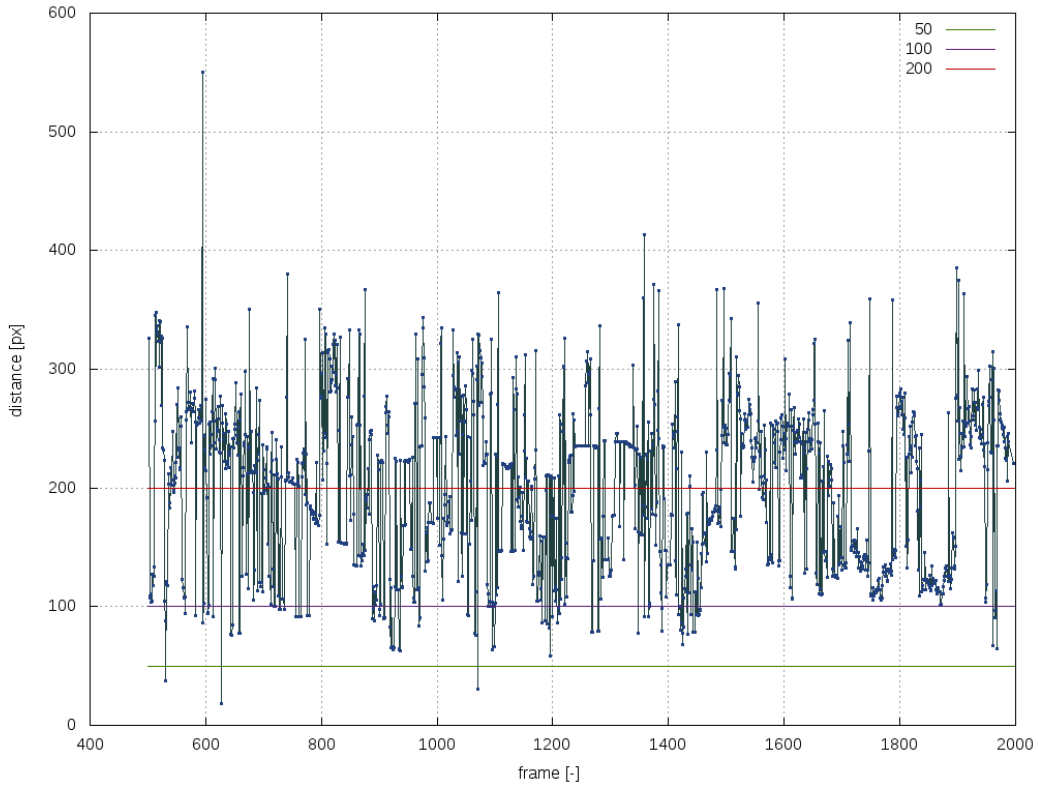

Figure 8: Distance of target 1 from nearest object in two views

[6] John A. Hartigan. *Clustering algorithms [by] John A. Hartigan.* Wiley New York,, 1975.

[7] Michal Hradiš and Roman Juránek. Real-time tracking of participants in meeting video. In *Proceedings of The 10th Central European Seminar on Computer Graphics*, page 5, 2006.

[8] Intel. *OpenCV (Open Source Computer Vision) Library*, 2007.

[9] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *In British Machine Vision Conference*, volume 1, pages 384–393, 2002.

[10] Jozef Mlích and Petr Chmelař. Trajectory classification based on hidden markov models. In *Proceedings of 18th International*

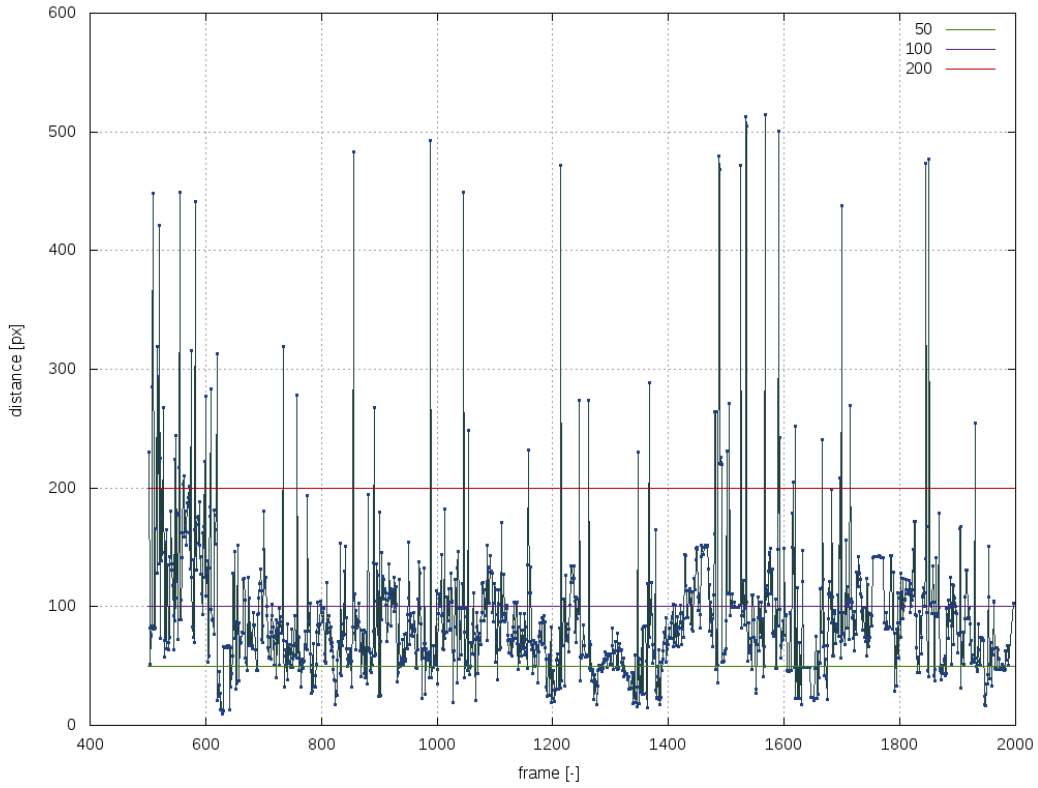

Figure 9: Distance of target 2 from nearest object in two views

Conference on Computer Graphics and Vision, pages 101–105. Lomonosov Moscow State University, 2008.

[11] Nuria M. Oliver, Barbara Rosario, and Alex P. Pentland. A bayesian computer vision system for modeling human interactions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):831–843, 2000.

[12] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image Processing, Analysis, and Machine Vision*. Thomson-Engineering, 2007.

[13] Tsai R. Y. An efficient and accurate camera calibration technique for 3d machine vision. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1986.

distance [px]	t0	t1	t2
5	3.6	0.0	2.2
10	12.4	0.2	6.8
20	37.5	0.6	21.7
30	53.9	02.8	44.0
40	64.7	09.2	59.6
50	73.9	15.8	70.8
75	90.5	36.6	88.2
100	97.2	67.2	92.0
150	99.6	99.1	96.3
200	99.8	99.8	97.7

Table 1: 2d table

[14] Zhengyou Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *in ICCV*, pages 666–673, 1999.

[15] Qiang Zhu, Shai Avidan, and Kwang-Ting Cheng. Learning a sparse, corner-based representation for time-varying background modeling. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pages 678–685, Washington, DC, USA, 2005. IEEE Computer Society.

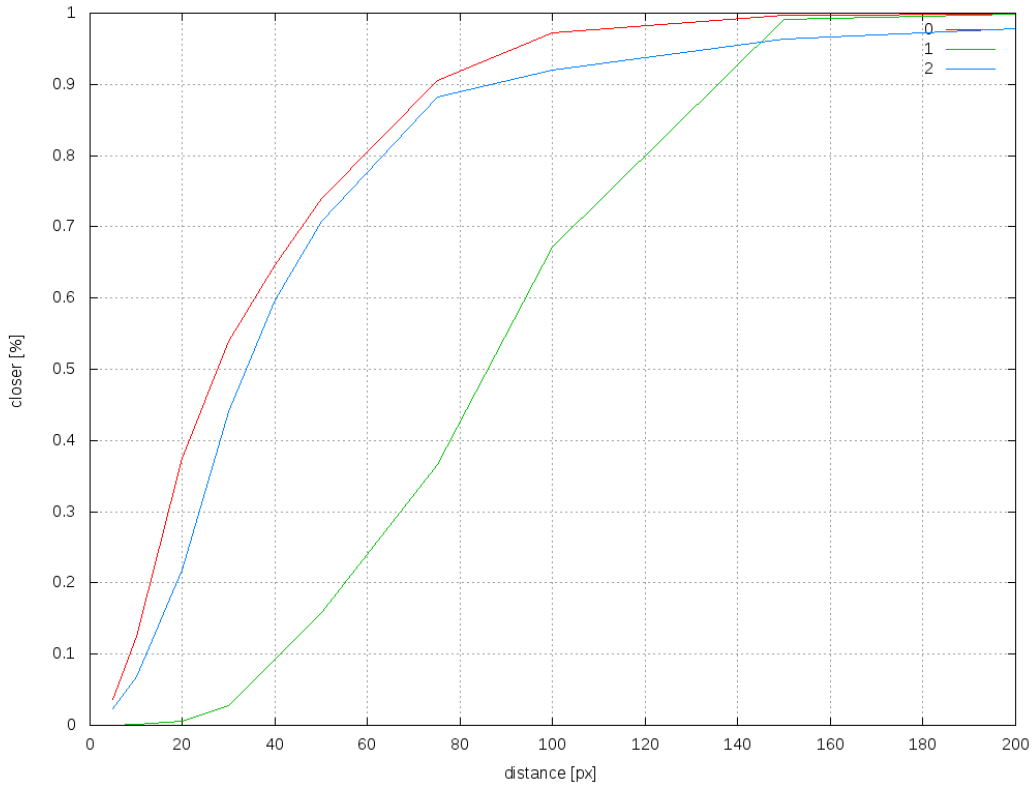

Figure 10: Distance in 2D

A Smart meeting room - additional object dimensions

From: Olivier Masson <olivier.masson@idiap.ch>

Hi Joseph,

We mooved the Smart Meeting Room. Some values are not accessible any more:

- position of pictures on the wall
- position of the whiteboard

However some heights/dimensions are possible:

- height of picture on the wall: 70 cm

distance [px]	t0	t1	t2
5	0.0	0.0	0.0
10	0.0	0.0	0.0
20	0.0	0.0	1.2
30	0.0	0.0	4.7
40	0.8	0.2	8.4
50	4.5	0.2	18.2
75	14.8	1.2	46.3
100	17.1	6.5	69.4
150	18.0	31.1	91.2
200	21.1	45.8	96.0
250	25.1	77.0	97.4
300	25.1	93.8	98.5
350	26.2	98.7	98.7
400	41.9	99.8	98.7
500	78.4	99.9	99.7

Table 2: Distance in 4D

- height of the table: 74 cm
- height of the bookshelf on the right: 203 cm

I can give you some extra dimension if you want:

- whiteboard: 90cm x 120 cm
- screen display (white part): 142cm x 190cm

Olivier

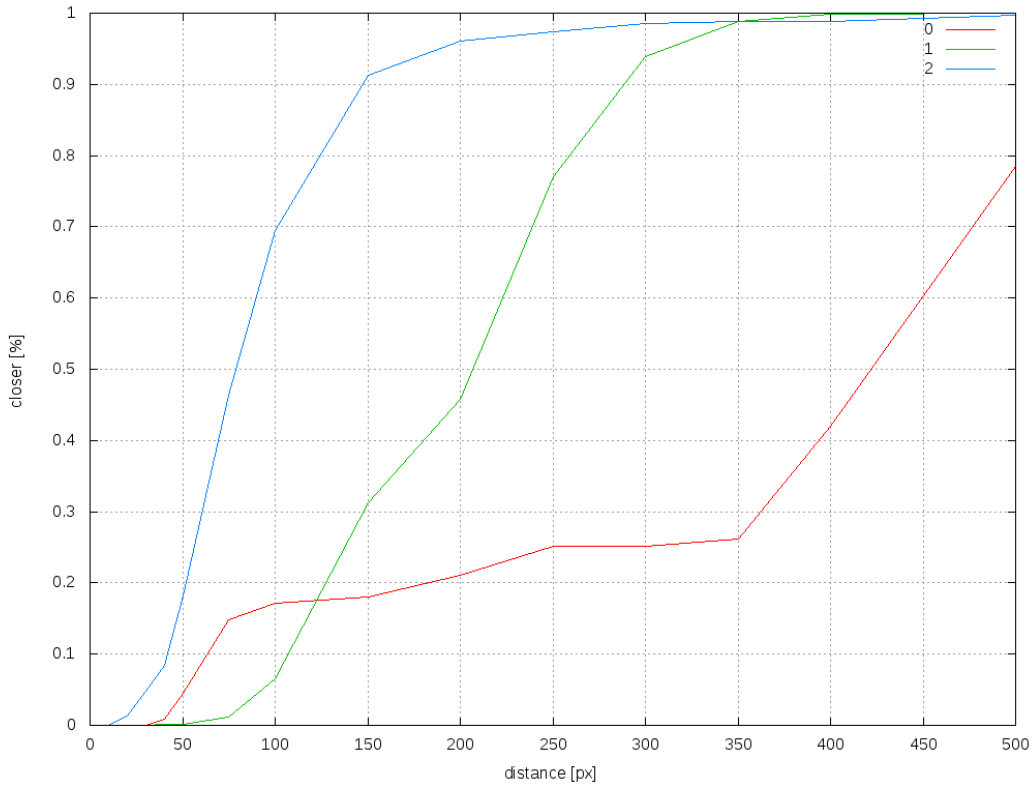

Figure 11

B IDIAP Smart meeting room setup

The calibration data are related to the first frame of `IS1000a-C.avi` and `IS1000a-L.avi`.

World Coordinates [mm]			Camera C [px]		Camera L [px]	
x	y	z	x	y	x	y
0	0	0	165	369		
3550	0	0	628	378		
3550	2250	0	644	21		
0	2250	0	147	25		
3130	0	1040	620	451		
3130	0	1840	665	532		
1740	1060	2230	378	322	345	470
3130	1807.83	1040	649	103	30	66
3130	1425.46	1040	647	177	35	170
3130	1105.05	1040	645	239	40	262
3130	1105.05	1840	711	269	254	259
3130	540.88	1260.4	648	354	95	416
3130	540.88	1840	692	399	260	418
3550	1750	2230			345	75
3130	1425.46	1840			251	166
3130	1807.83	1840			250	56
3130	540.88	2640			478	416
3130	540.88	4480			676	411
3550	1800	1840			58	65
3550	1800	1840			260	249
3550	1807.83	1040			467	384
3550	1807.83	1840			258	52
3130	540.88	1040			55	415
3550	540.88	4480			651	377
2340	740	3030	609	566	626	475
1740	740	3030	359	572		
1140	740	3030	123	547		
1140	740	1430	261	337		
2340	740	1430	514	340	65	477
1740	740	2230	379	414		
3130	2030	1040	649	60	28	4
3130	2030	3440			701	3

Table 3: World and image coordinates for IDIAP smart meeting room