

Optical Tracking II



Research project

Emanuel Züger, Uwe Hahne

Overview



- Introduction
- Frustum reconstruction
- Low-cost optical tracking
- Conclusion and future work

Introduction



- Frustum reconstruction
 - ART
 - IR-beamer
- Low-cost optical tracking
 - webcams
 - cheap objects

Introduction

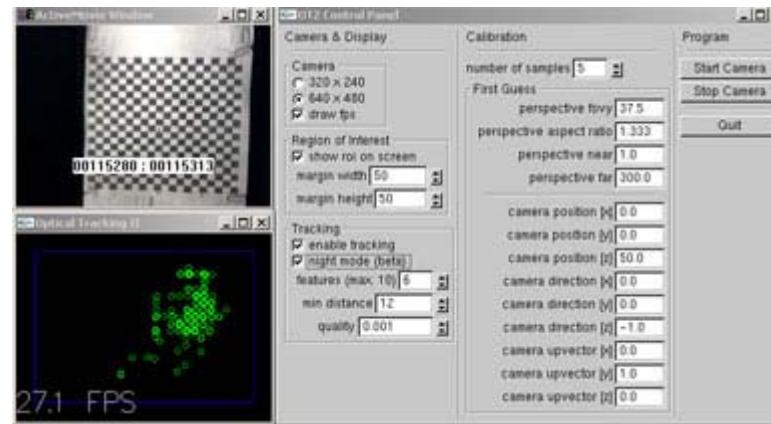


- Prior work
 - Optical tracking 1
 - Multi camera approach
 - ARToolkit
 - LED-Tracking

Frustum reconstruction



- Image processing with OpenCV
- Laserpoint tracking
- Camera calibration
- Building a lookup-table



Frustum reconstruction

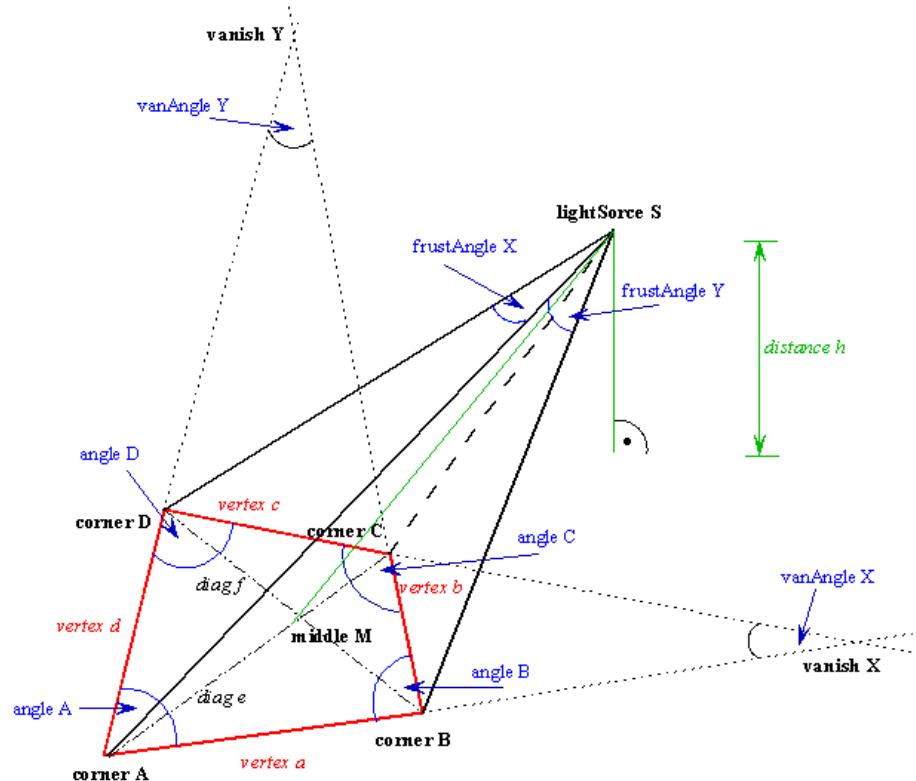


- head/hand tracking system
- Using a rack of IR-emitters
- Reconstruction out of the deformed figure

Frustum reconstruction



- First analytical approach
 - The angle of the opposing edges is same as the difference to the perpendicular



Frustum reconstruction



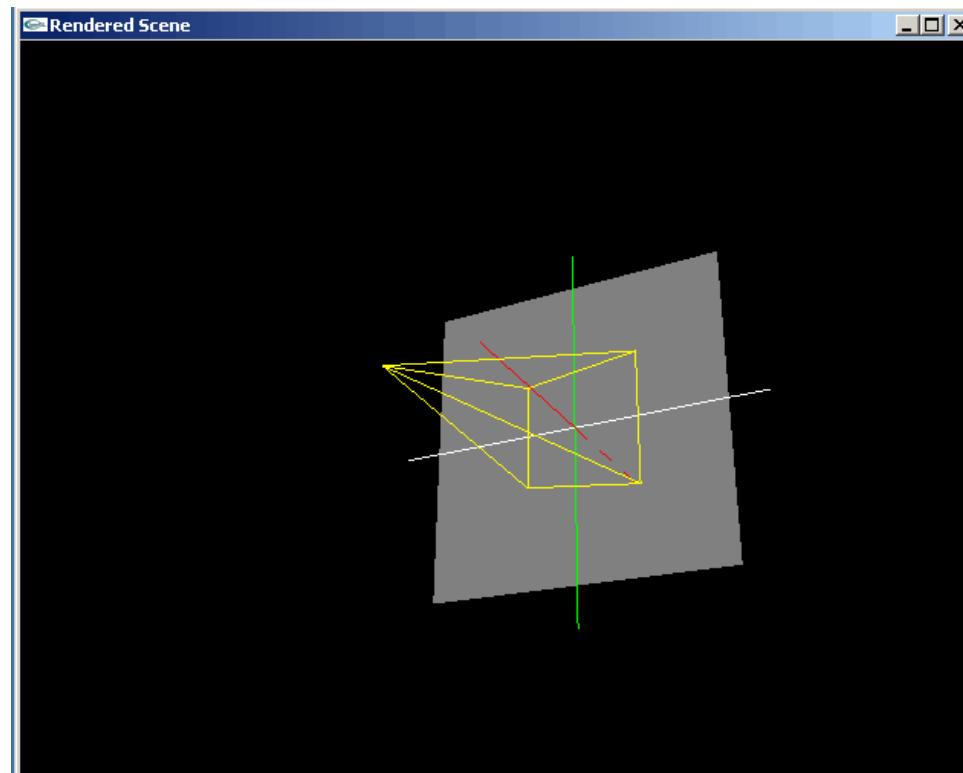
- Second analytical approach
 - Building a equation system from the ratio of edges and angles of the pyramid
 - worked but unsolvable

```
Maple 6 - [pyramide8.mws]
File Edit View Insert Format Spreadsheet Options Window Help
[Icons]
x [Close] [Minimize] [Maximize]
[sx, p1x, p1y, p2x, p2y, s1x, s1y, s2x, s2y, alpha]
2*sx*p2x*p1x*p3x+24*sx*p1x*p2y*2*p2x*p3x+6*p2y*2*p1y*2*p2x*2*p3x*2+6*p2x*4*p1x*2*p3x*2-cos(alpha)^4*p1y^4*p2y^2*p3y^2+24*p2y*2*p1y*p2x*2*p3x*p1x*p3y-12*p2y*p1y*p2x*3*p3x*2*p1x-12*sx^3*p1x*p2y*p3x^2;
sy = RootOf(6*sx^4*p1x^2*p3x^2+6*p2y^4*p3y^2-4*sx^4*p1x*p3x^3+p2y^4*p3y^4+36*sx*p1x*p2y*p3y*p2x^2*p3x^2-36*p2x^2*p1x^2*p2y*p3y*sx*p3x-12*p2x^2*p1x*p2y^2*p3y-12*p2x^3*p1x*p2y*p3y*p2x^2+p2x^4*p3x^4-4*p2x*sx^3*p3x^4-4*p2y^4*p3y^3*p3y-4*p2x^3*p1x^4*sx-4*p2y^4*p3y^3*p3y+6*sx^2*p1x^2*p2y^2*p3y^2-24*p2x^2*sx^2*p3x^2*p1x+16*p2x*sx^3*p3x^3*p1x+36*sx^2*p2x^2*p1x^2*p3x^2-24*sx^3*p2x*p1x^2*p3x^2+16*sx*p2x^3*p1x*p3x^3-4*sx^3*p3x^3*p2y*p3y+4*p2x^3*p3x^3*p2y*p3y+12*p2x*p1x*p2y^3*p1y^2*p3y+sx^4*p3x^4-4*p2x^3*sx*p3x^4+24*sx*p1x^2*p2y^2*p3y*p2x*p3y+36*sx^2*p1x*p2y*p3y*p2x*p3x^2-36*sx^2*p1x^2*p2y*p3y*p2x*sx^2*p3x^2*p2y*p3y-12*p2x^2*sx*p3x^2*p2y*p3y-4*p2y*p3y*p2x^3*p3x^3-36*sx*p1x*p2y*p3y*p2x^2*p3x^2+12*sx*p1x*p2y^3*p1y^2*p3y+24*sx^2*p1x*p2y^2*p3y*p3x*p3y-48*sx*p1x*p2y^2*p3y*p2x*p3x*p3y-12*p2y^2*p3y*p2x^2*p3x^2*p3y+12*p2y^3*p1y^2*p2y*p3x*p3y-12*sx*p1x*p2y^2*p3y^2*p2x-36*sx^2*p1x*p2y*p3y*p2x*p3x^2+6*p2x^2*sx^2*p3x^4-12*sx^2*p1y*p2y*p3x^3*p2x-12*sx^2*p1y*p2y^2*p3x^2*p3y+24*sx*p1y*p2y^2*p3x^2*p2x*p3y+12*sx*p1y*p2y*p3x^3*p2x^2-12*sx*p1y^2*p2y^3*p3x*p3y+36*sx^2*p1x^2*p2y*p3y*p2x*p3x-12*sx^3*p1x^2*p2y*p3y*p3x+12*sx^3*p1x*p2y*p3y*p3x^2+4*p2y^4*p3y^4*cos(alpha)^4-5*p2y^4*p3y^4*cos(alpha)^2+4*p2x^4*p3y^4*cos(alpha)^4-4*p2x^4*p3y^4*cos(alpha)^6+6*sx^2*p1y^2*p2y^2*p3x^2-cos(alpha)^4*p2x^4*p1x^4+2*cos(alpha)^6*sx^2*p2y^6-4*p2y^6*p3y^2*cos(alpha)^6+4*p2y^5*p3y^3*cos(alpha)^2+2*sx^5*p3x^3*cos(alpha)^2+13*p2x^4*sx^4*cos(alpha)^4-6*p2x^5*sx^3*cos(alpha)^4+4*cos(alpha)^4*sx^6*p2x^2+4*cos(alpha)^4*p1x^4*p2y^4-12*cos(alpha)^4*sx^5*p2x^3-2*cos(alpha)^6*sx^4*p2y^4+4*cos(alpha)^6*p2y^4*p3y^4-sx^4*p3x^4*cos(alpha)^2-cos(alpha)^2*p2x^4*p1x^4-2*cos(alpha)^2*sx^5*p1x^3+2*cos(alpha)^6*p2y^6*p3y^2-5*cos(alpha)^2*p2y^4*p3y^4-2*cos(alpha)^6*p2y^4*p3y^4+2*cos(alpha)^2*p2y^5*p3y^3+2*cos(alpha)^6*p1x^2*p2y^6+cos(alpha)^4*sx^4*p1y^4-p2x^4*p3x^4*cos(alpha)^2+2*p2x^5*p1x^3*cos(alpha)^2+2*p2x^6*p1x^2*cos(alpha)^4+cos(alpha)^2*sx^4*p1x^4+36*sx*p1x^2*p2y*p3y*p2x^2*p3x+cos(alpha)^4*sx^4*p2y^4-12*sx*p1y^2*p2y^2*p3x^2*p2x+12*p2x^3*p1x^2*p2y*p3y*p3x-4*p2x^4*p3x^3*p1x-24*sx*p2x^3*p1x^2*p3x^2-12*p2y^3*p3y^2*sx*p1x*p3y+2*cos(alpha)^4*p2y^6*p3y^2-cos(alpha)^4*p2y^6*p3y^2+12*p2y^3*p3y^2*sx*p1y*p3x-12*sx^2*p1x*p2y^2*p3y^2*p3x-4*sx^3*p1x^3*p2y*p3y-4*sx*p1x*p2y^3*p3y^3+4*sx*p1y^3*p2y^3*p3x-4*p2y^3*p3y^3*p2x*p3x)
```

Frustum reconstruction



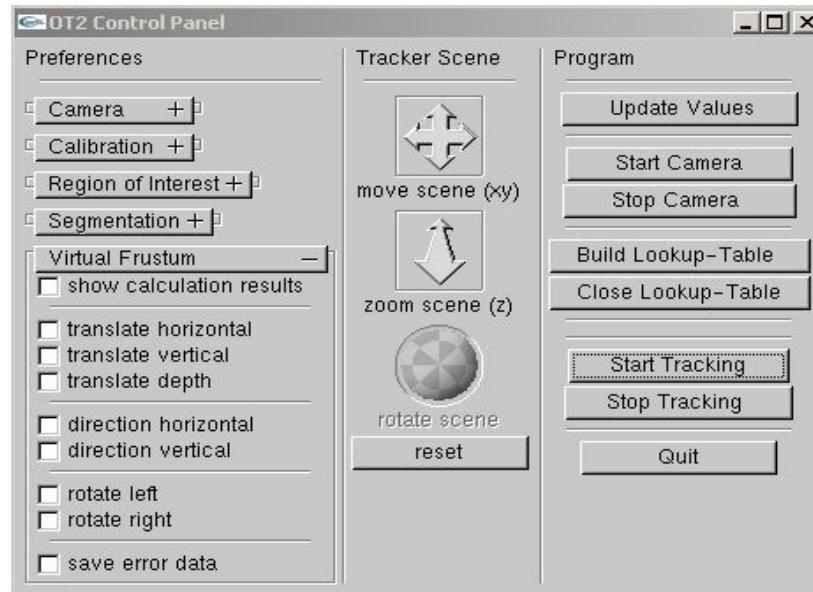
- Succeeded approach
 - Numerical solution
 - Powell's direction set method



Frustum reconstruction



- demonstration





Low-cost optical tracking

Low-cost tracking



- cheap and simple components
 - webcams (Logitech Quickcam)
 - simple tracking devices
(colored ball, fingertip,
etc.)

Low-cost tracking

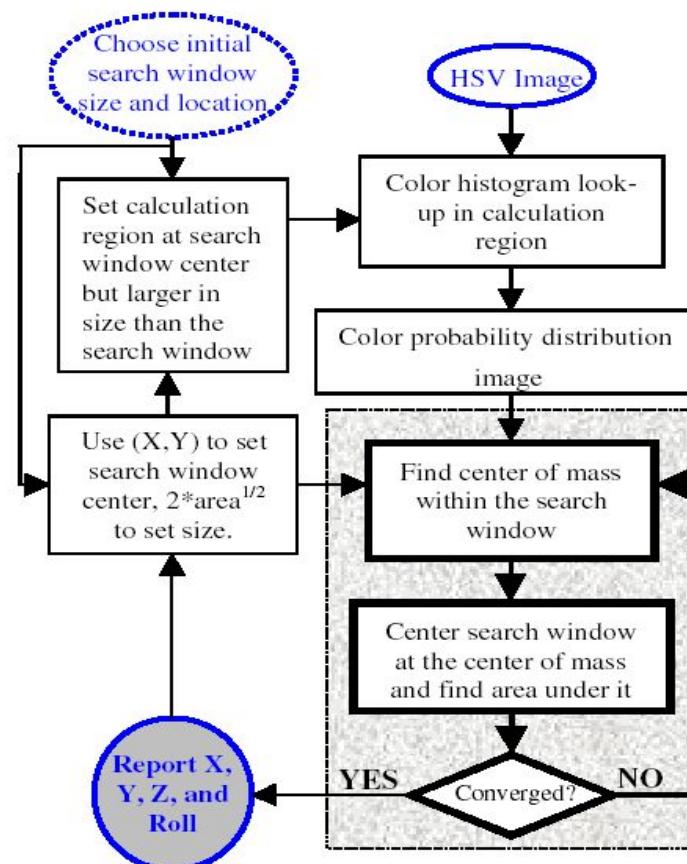
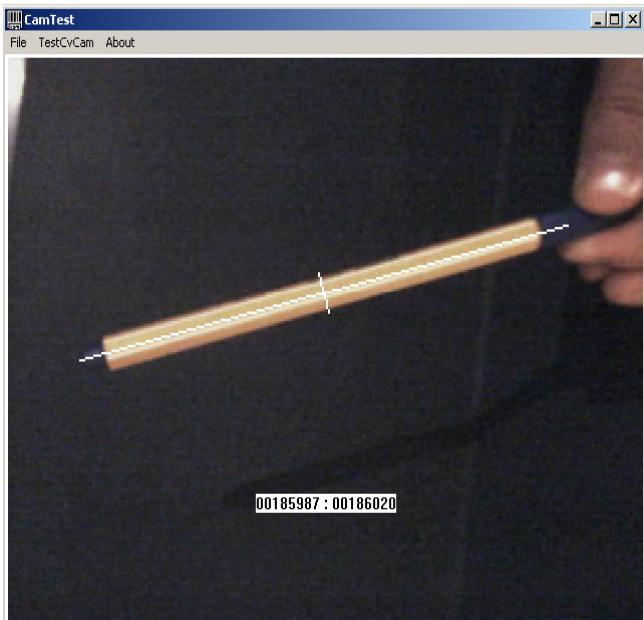


- Image processing
 - recognizing objects in the video-stream
 - critical part: accuracy depends of good segmentation

Low-cost tracking



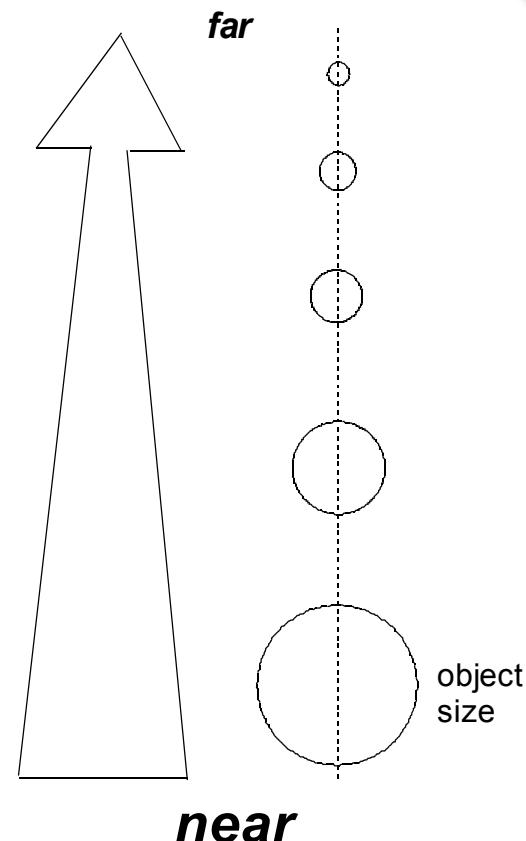
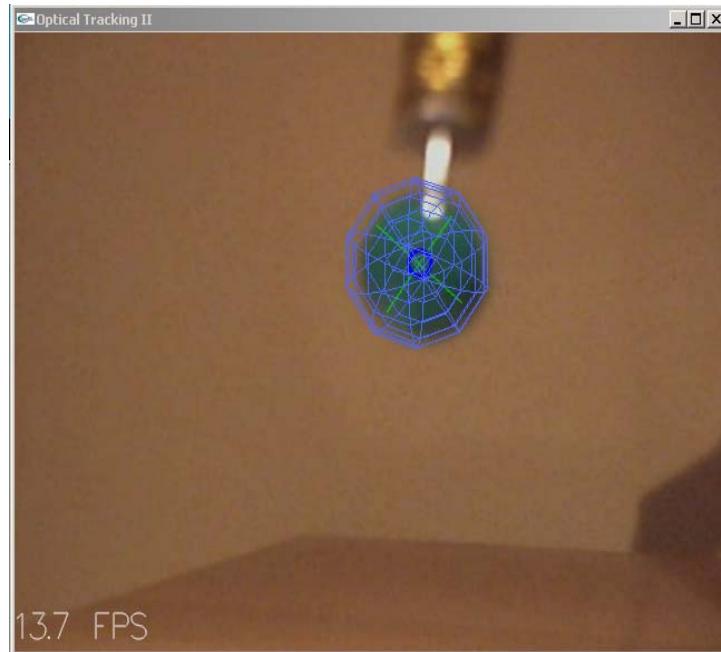
- Camshift algorithm



Low-cost tracking



- Single camera approach



Low-cost tracking

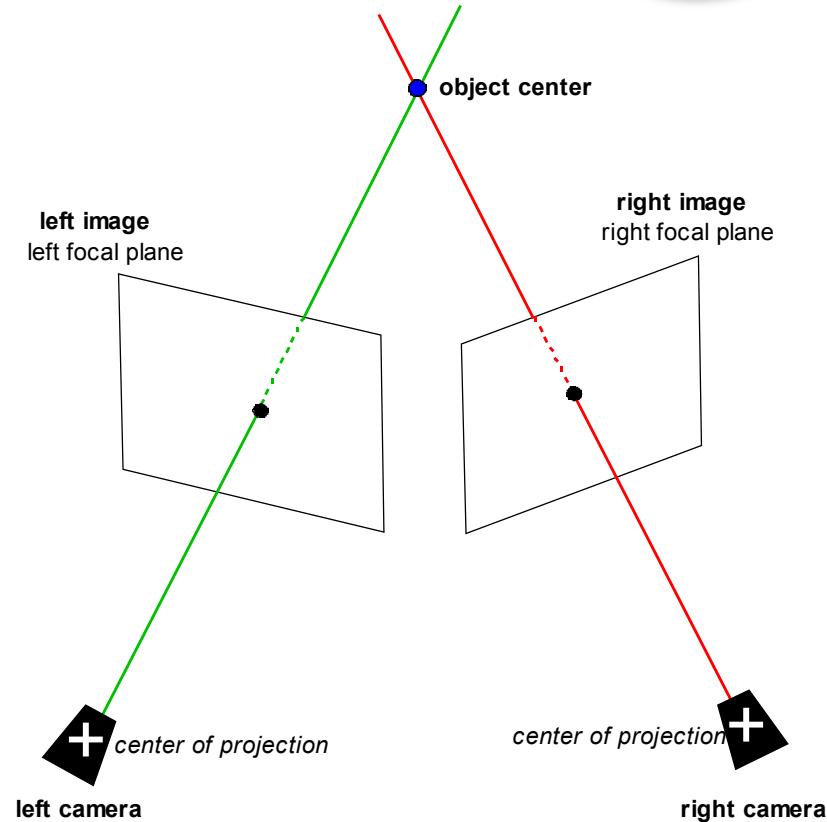


- Draw-back single camera
 - object's size critical factor
 - camera resolution too low
 - deviation:
 - ~ 10 mm (10-20 cm distance)

Low-cost tracking



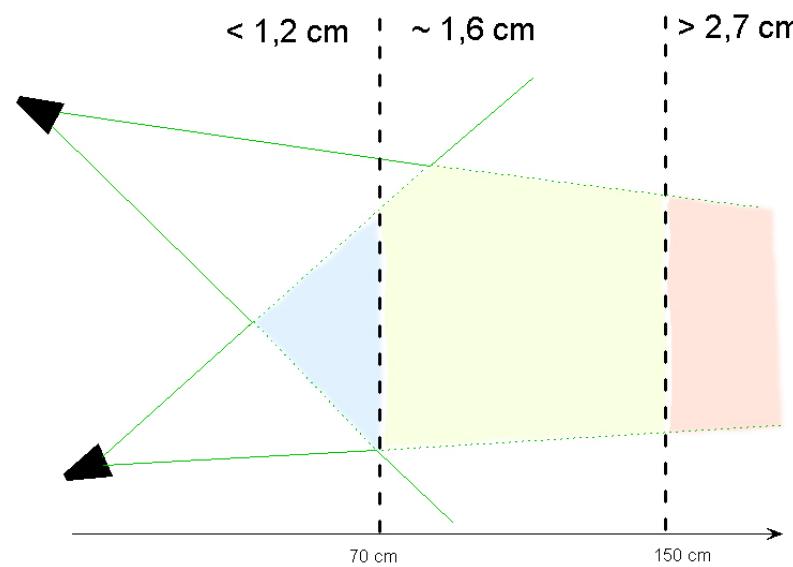
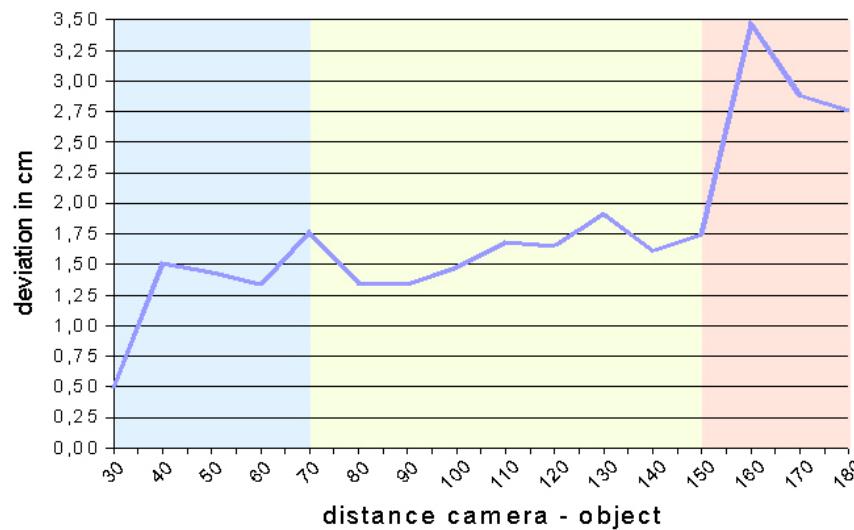
- Stereo camera approach
 - two computers
 - data exchange via TCP/IP
 - Stereo photogrammetry (triangulation)



Low-cost tracking



- Results
 - Accuracy:



Low-cost tracking



- Results

- Performance
- More hardware needed
- restricted tracking-area

Objects	Fps
0	115
1	30
2	18
3	12
4	9, 5

*ATHLON XP
1800+*

1024 Mb RAM

*GeForce4 Ti
4600*



Conclusion and Future work

Conclusion & Future work



- Three different approaches
 - efficiency
 - accuracy
 - complexity of calibration and installing

Conclusion & Future work



- Possible applications
 - Commercial use (e.g. ART)
 - Cooperation with warhol-shader project
 - Input device for the virtual Showcase

Conclusion & Future work



- Suggestions for next project
 - Image processing: using other algorithms
 - Experiments with other tracking-devices



End of presentation