Parallel tracking and mapping

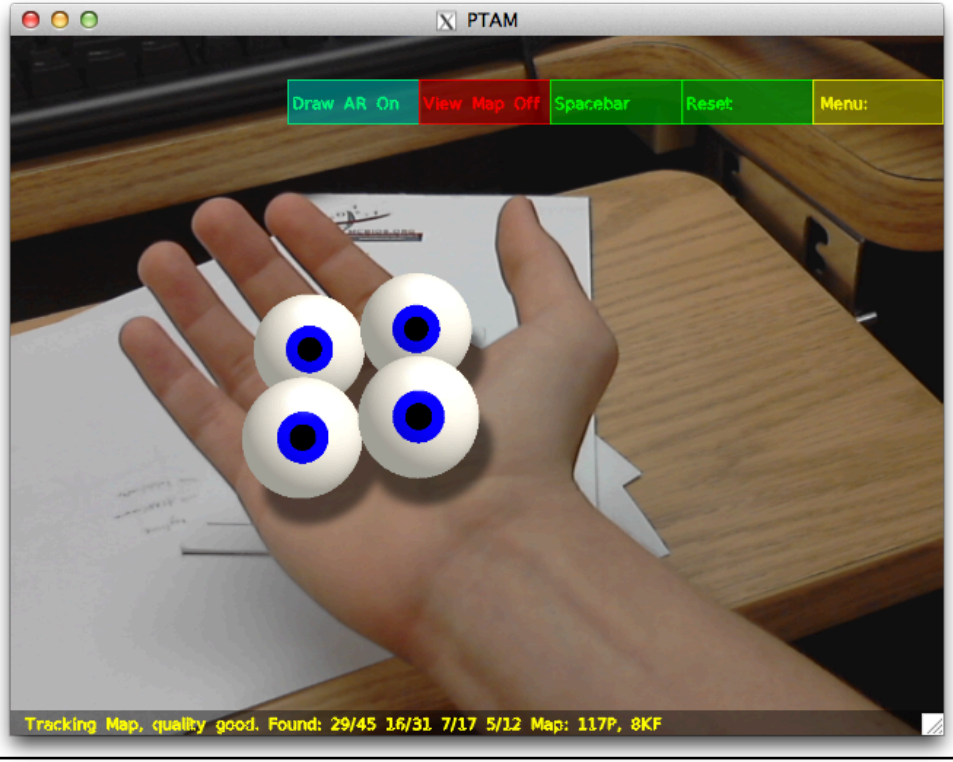
Michael Macias

2013-04-30

Have you ever wanted to hold a pair of eyes in your hand but don’t have any to spare?

With augmented reality, now you can!

Augmented reality

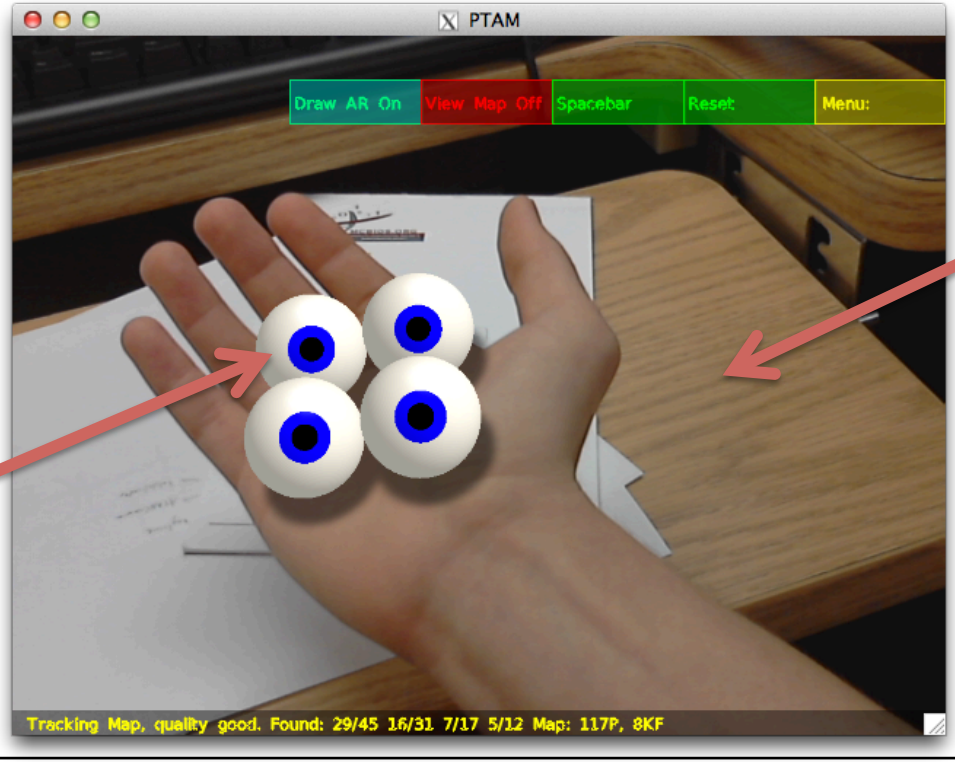


aug·ment·ed re·al·i·ty

an artificial environment created through the combination of real-world and computer-generated data

-- Dictionary.com

Augmented reality



real-world

computer-generated

Augmented reality

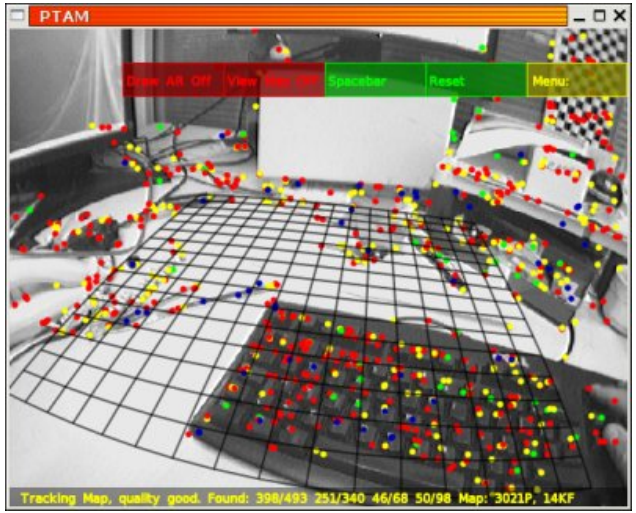
Tons of room to improve performance and capability

Georg Klein

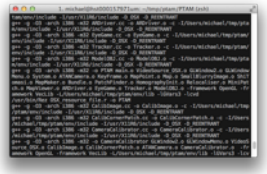
David Murray

PTAM

?!


PTAM
Spacebar
Reset
Menu:
Tracking Map, quality good. Found: 398/493 251/340 46/68 50/98 Map: 3021P, 14KF

Project timeline



compile code

map paper
to code

custom AR
model

libCVD
⇔
OpenCV

Compiling

Trivial on Ubuntu 8.04.4 LTS

Compiling on Ubuntu

```
michael:PTAM $ sudo apt-get update
michael:PTAM $ sudo apt-get install $DEPS_DEV
michael:PTAM $ for dep in deps/*; do
  cd $dep
  ./configure && make
  cd ..
done
michael:PTAM $ make
```

Compiling

Nontrivial on OS X

Compiling on OSX

Project timeline

compile code

map paper
to code

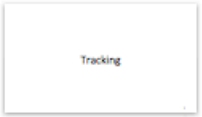
custom AR
model

libCVD
⇔
OpenCV

Map paper to code

1


2



3

4

5

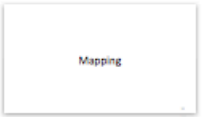
6

7

8

9

10



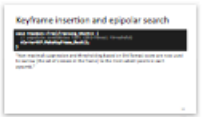
11

12

13

14

15



16

17

18

19

Project timeline

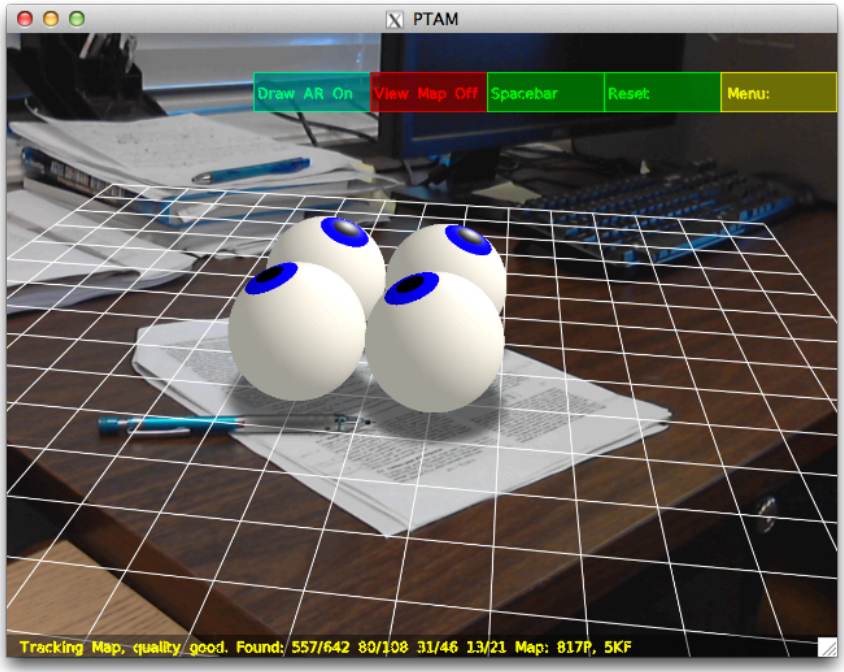
compile code

map paper to code

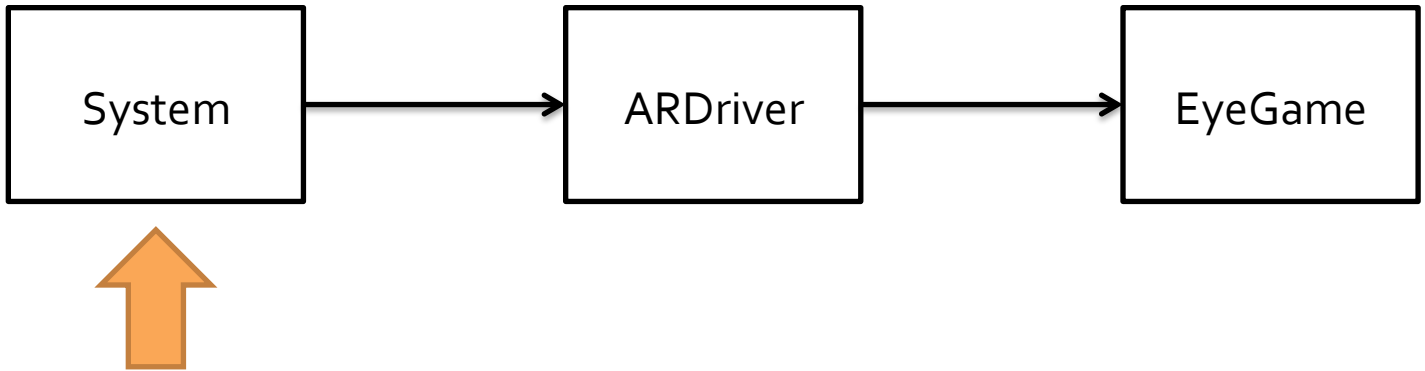
custom AR model

libCVD ⇔ OpenCV

AR in PTAM


PTAM
Draw AR On
View Map Off
Spacebar
Reset
Menu:
Tracking Map, quality good. Found: 557/642 80/108 31/46 13/21 Map: 817P, 5KF

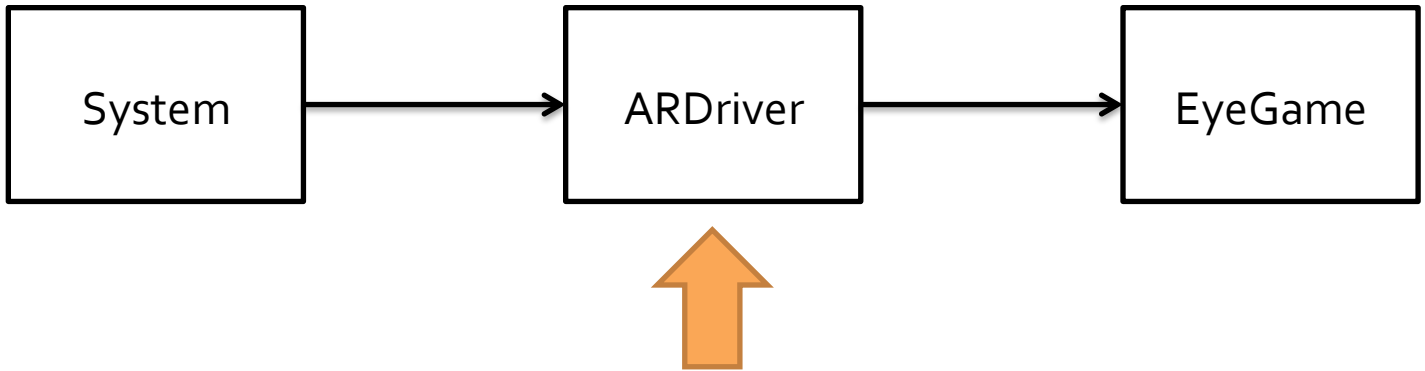
AR in PTAM



System.cpp

```
void System::Run() {
  while(!mbDone) {
    // ...
    if(bDrawMap) {
      mpMapViewer->DrawMap(mpTracker->GetCurrentPose());
    } else if (bDrawAR) {
      mpARDriver->Render(mimFrameRGB, mpTracker->GetCurrentPose());
    }
    // ...
  }
}
```

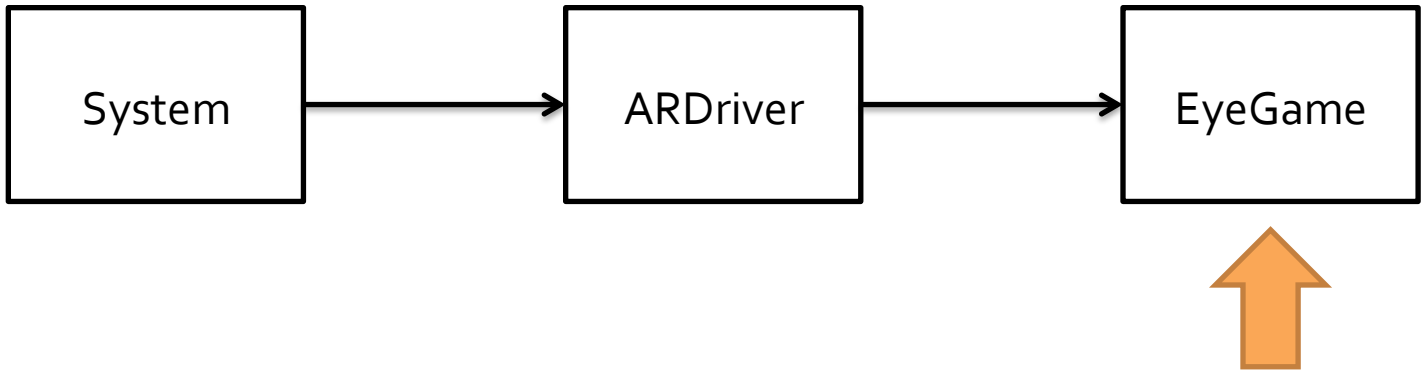
AR in PTAM



ARDriver.cpp

```
void ARDriver::Render(Image<Rgb<byte> > &imFrame, SE3<> se3CfromW) {
  // ...

  mGame.DrawStuff(se3CfromW.inverse().get_translation());

  // ...
}
```

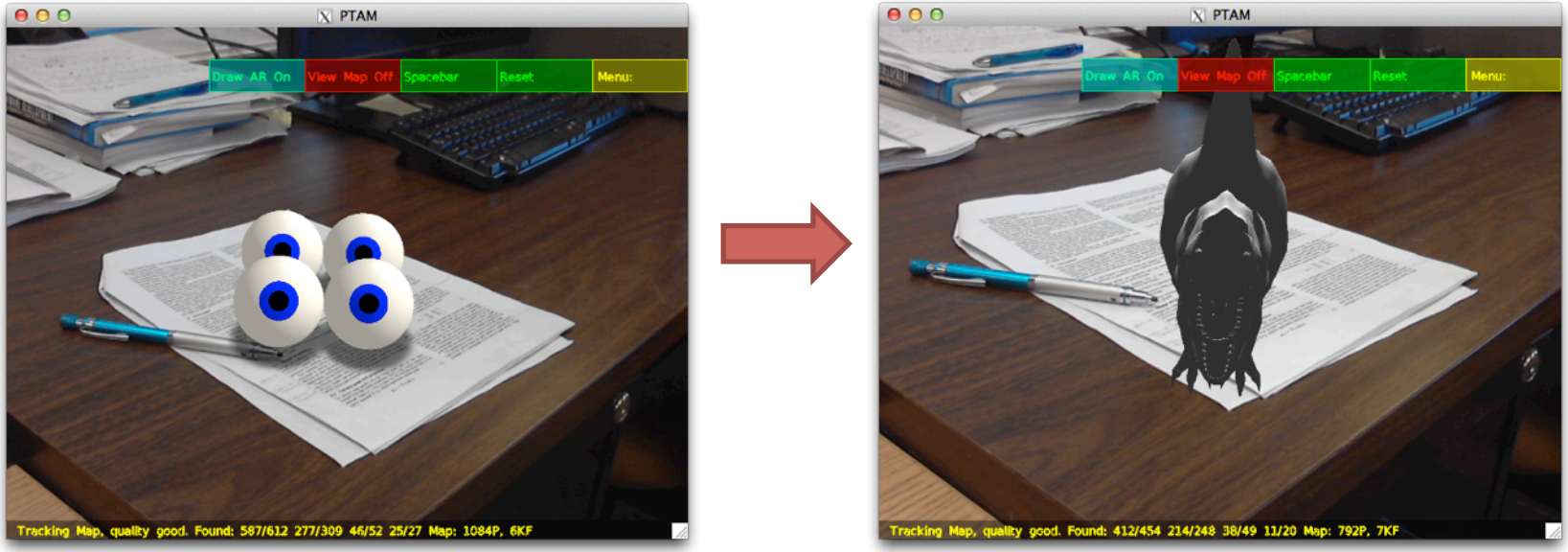
AR in PTAM



EyeGame.cpp

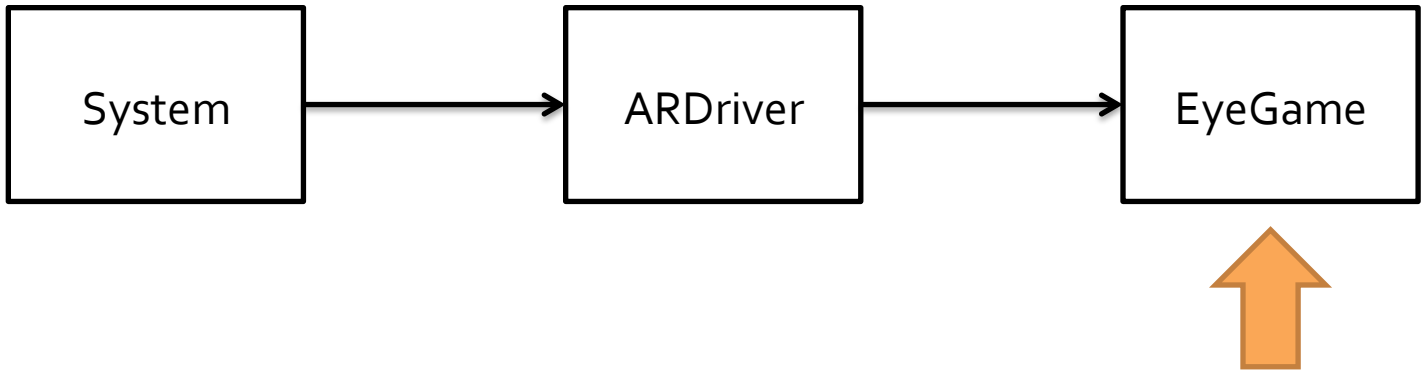
```
void EyeGame::DrawStuff(Vector<3> v3CameraPos) {
  // ...
  LookAt(i, v3CameraPos, 0.02);
  // ...
  glBegin(GL_TRIANGLE_FAN);

  for(int i = 0; i < nSegments; i++) {
    // add verticies
  }

  glEnd();
  // ...
}
```

Importing custom models




Blender
File
Add
Render
Window
Help
Default
Scene
Blender Render
v2.66.1 | Verts:5550 | Faces:10876 | Tris:10886 | Objects:1/5 | Lamps:0/1 | Mem:15.41M (5.68M
New Cmd N
Open... Cmd O
Open Recent... Shift Cmd O
Recover Last Session
Recover Auto Save...
Save Cmd S
Save As... Shift Cmd S
Save Copy... Ctrl Alt S
User Preferences... Cmd ,
Save Startup File Ctrl U
Load Factory Settings
Link Ctrl Alt O
Append Shift F1
Import
Export
External Data
Quit Cmd Q
Collada (Default) (.dae)
Motion Capture (.bvh)
Stanford (.ply)
Stl (.stl)
3D Studio (.3ds)
Autodesk FBX (.fbx)
Wavefront (.obj)
X3D Extensible 3D (.x3d)
Camera Persp
(1) Trex_Rigged
Right Ortho
Mirror Editing
Proportional Editing
Disable
Proportional Editing Fall
Smooth
Proportional Size
1.000
Edit Texture Space
Confirm on Release
View
Select
Object
Object Mode
Global
View
Marker
Frame
Playback
Start: 1
End: 250
No Sync
OBJ_MuseumEx
Preview
Diffuse
Lambert
Specular
CookTorr
Hardness: 3
Shading

AR in PTAM

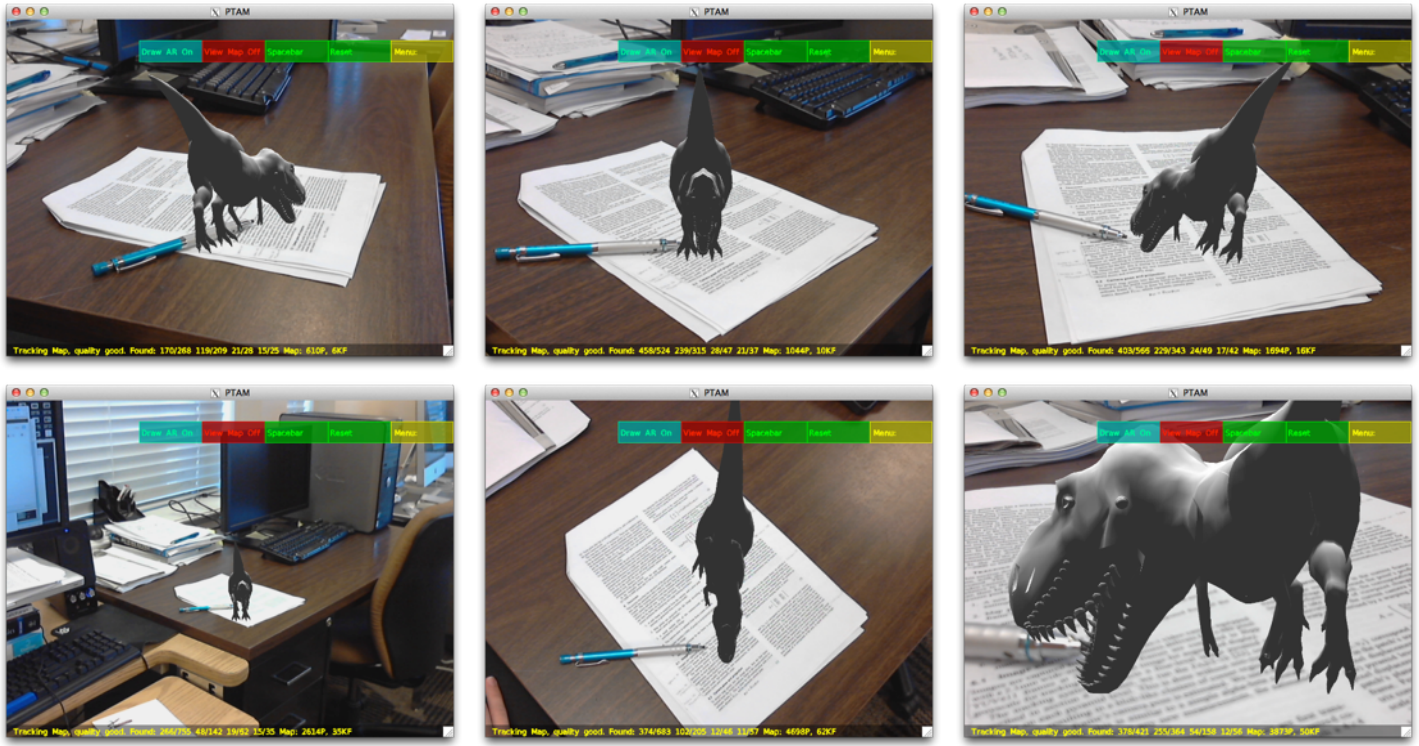


EyeGame.cpp

```
void EyeGame::DrawStuff(Vector<3> v3CameraPos) {
  glMatrixMode(GL_MODELVIEW);
  glLoadIdentity();
  glRotatef(90.0f, 1.0f, 0.0f, 0.0f);
  glRotatef(180.0f, 0.0f, 1.0f, 0.0f);
  // get model
  glVertexPointer(3, GL_FLOAT, nVerticies, verticies);
  glDrawElements(GL_TRIANGLES, nTriangles * 3, GL_UNSIGNED_INT,
      indexBuffer + startIndex);
}
```

Demo time!

Backup results



Project timeline

compile code

map paper to code

custom AR model

libCVD
⇔
OpenCV

Camera pose estimation

CVD

libCVD

✓ camera input
✓ camera calibrator
✗ camera pose estimator
…

OpenCV

✓ camera input
✓ camera calibrator
✓ camera pose estimator
…

Camera pose estimation

More to be determined…

References

- Klein, Georg, and David Murray. "Parallel tracking and mapping for small AR workspaces." *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*. IEEE, 2007.
- Parallel Tracking and Mapping for Small AR Workspaces (PTAM). <**http://www.robots.ox.ac.uk/~gk/PTAM/**>.
- OpenGL – The Industry Standard for High Performance Graphics. <**http://www.opengl.org/**>.
- OpenCV. <**http://opencv.org/**>.

Questions?