Efficient and Robust Circle Grids for Fiducial Detection

Master’s Colloquium

Michael Loipführer

Department of Informatics

May 23, 2022
What are fiducial markers?

- ARToolkitPlus [1]
- AprilTag [2], [3]
- ArUco [4]
- STag [5]
- RuneTag [6]
- Pi-Tag [7]
- TopoTag [8]
- LFTag [9]
What are fiducial markers?

1. AprilTag website. URL: https://april.eecs.umich.edu/software/apriltag (visited on 05/16/2022).
2. Researchgate. URL: https://www.researchgate.net/figure/Circular-marker-for-Augmented-Reality_fig1_259752709 (visited on 05/21/2022).
What are fiducial markers?

Designed for

- robust detections
- stable 3D pose recovery
- unique marker identifiers
Motivation and Goals

Motivation

Existing methods do not work well in very distorted images
Motivation and Goals

Motivation
Existing methods do not work well in very distorted images

Goals of this Thesis
Design a novel fiducial marker system based on circle grids that specifically

- provides robust detections
- achieves stable pose recovery
- works well with distorted fisheye cameras
- can be composed into larger marker grids
Outline

1. Introduction
2. Marker Design
3. Marker Grids
4. Evaluation
5. Live Demo
6. References
Marker Design

Circles
- robust detections even with high distortion
- subpixel accuracy

3 × 3 Grid
- compact layout
- regular geometry helps detection scheme

Example Marker
Marker Design

Marker Payload
- recover the orientation of a tag
- discard erroneous detections

Corners: 0111, Full payload: 021110121
Marker Design

Marker Payload
- recover the orientation of a tag
- discard erroneous detections

Coding Scheme
- Lyndon words for rotational uniqueness in corners
- remaining 5 digits for arbitrary error detection

Corners: 0111, Full payload: 021110121
Detection Pipeline

Original image
Thresholded image
Ellipse candidates
Detected ellipses
Marker candidates
Valid markers
Pose Estimation
Marker Candidate Generation
Marker Candidate Generation

Basic Idea

- perform a KNN-search for every ellipse
- check if the 8 nearest neighbors fit a $3 \times 3$ grid
Marker Candidate Generation

Basic Idea

- perform a KNN-search for every ellipse
- check if the 8 nearest neighbors fit a $3 \times 3$ grid
- KNN-search with $k \geq 8$
- geometric heuristics based on angles and distances between ellipses to filter out invalid marker geometries
Payload Decoding

Ellipse Type Decoding
- relate relative distances between ellipses with measured sizes
- compare to known true sizes of different ellipse types

Payload Check
- recovery of rotation using payload of corner points
- validity check of any error detection schemes
Pose Estimation

Method
- all 9 ellipses used as 2D - 3D point correspondences
- UPNP [12] and KNEIP [13] used as initializations
- non-linear optimization for pose refinement
Pose Estimation

Method
- all 9 ellipses used as 2D - 3D point correspondences
- UPNP [12] and KNEIP [13] used as initializations
- non-linear optimization for pose refinement

Benefits of our method
- ellipse detection produces accurate 2D feature points
- more correspondences leads to more stable poses
Composable Marker Grids

AprilGrid
Composable Marker Grids

AprilGrid

Ours
Composable Marker Grids

Camera Calibration
- good performance for fisheye cameras
- more robust to detection errors than AprilGrid

Grids as Fiducials
- large number of unique marker identifiers
- more options for error detection and correction
Evaluation

Evaluation of
- detection precision and recall
- pose estimation accuracy
- in real world and simulated images
Evaluation - Simulated Data

Detection Recall

Pose Translation Error
# Evaluation - Real World Data

<table>
<thead>
<tr>
<th>Marker System</th>
<th>distance</th>
<th>low angle close</th>
<th>pan tilt close</th>
<th>mixed</th>
<th>planar close</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArUco</td>
<td>56.35%</td>
<td>83.37%</td>
<td>55.65%</td>
<td>89.78%</td>
<td>54.88%</td>
</tr>
<tr>
<td>STag</td>
<td>37.88%</td>
<td>63.00%</td>
<td>60.58%</td>
<td>86.69%</td>
<td>62.80%</td>
</tr>
<tr>
<td>Pi-Tag</td>
<td>10.38%</td>
<td>39.87%</td>
<td>7.68%</td>
<td>0.00%</td>
<td>0.26%</td>
</tr>
<tr>
<td>Ours</td>
<td>39.04%</td>
<td>82.34%</td>
<td><strong>71.01%</strong></td>
<td>88.11%</td>
<td><strong>66.23%</strong></td>
</tr>
</tbody>
</table>

**Detection Recall**

<table>
<thead>
<tr>
<th>Marker System</th>
<th>distance</th>
<th>low angle close</th>
<th>pan tilt close</th>
<th>mixed</th>
<th>planar close</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArUco</td>
<td>0.3334</td>
<td>0.0077</td>
<td>0.0458</td>
<td>0.2227</td>
<td>0.0236</td>
</tr>
<tr>
<td>STag</td>
<td>0.1969</td>
<td>0.1013</td>
<td>0.0850</td>
<td>0.1319</td>
<td>0.1204</td>
</tr>
<tr>
<td>Pi-Tag</td>
<td>0.0900</td>
<td>0.0212</td>
<td>0.0229</td>
<td>nan</td>
<td>nan</td>
</tr>
<tr>
<td>Ours</td>
<td><strong>0.0468</strong></td>
<td><strong>0.0024</strong></td>
<td><strong>0.0044</strong></td>
<td><strong>0.0325</strong></td>
<td><strong>0.0051</strong></td>
</tr>
</tbody>
</table>

**Root Mean Translation Error**

18 / 29
Evaluation - Qualitative Results

- Low viewing angle
- Large image distortion
- Large image distortion
- Significant motion blur
- Difficult lighting
- Large viewing distance
## Evaluation - Failure Cases

<table>
<thead>
<tr>
<th>Marker System</th>
<th>LabelMe</th>
<th>Indoor Scene Recognition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArUco</td>
<td>110 (3.77%)</td>
<td>193 (1.24%)</td>
</tr>
<tr>
<td>STag</td>
<td>293 (10.03%)</td>
<td>308 (1.97%)</td>
</tr>
<tr>
<td>Ours</td>
<td>2672 (91.51%)</td>
<td>3146 (20.15%)</td>
</tr>
</tbody>
</table>


Examples of false positive detections.
Live Demo

Questions?
References I


References II


[10] AprilTag website. URL: https://april.eecs.umich.edu/software/apriltag (visited on 05/16/2022).


Backup Slides
Marker Candidate Generation

Geometric Heuristic

- find closest neighbor to center ellipse
- perform heuristic filtering based on angle to the closest neighbor
Marker Candidate Generation

Geometric Heuristic

- find closest neighbor to center ellipse
- perform heuristic filtering based on angle to the closest neighbor
Marker Candidate Generation

Geometric Heuristic

- find closest neighbor to center ellipse
- perform heuristic filtering based on angle to the closest neighbor
Marker Candidate Generation

Geometric Heuristic

- find closest neighbor to center ellipse
- perform heuristic filtering based on angle to the closest neighbor
Marker Candidate Generation

Geometric Heuristic

- brute force search on the remaining neighboring ellipses for best alignment into $3 \times 3$ grid
## Composable Marker Grids - Grid Generation

<table>
<thead>
<tr>
<th>Grid size</th>
<th>Runtime</th>
<th>Number of grids</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 × 4</td>
<td>2.64s</td>
<td>262,144</td>
</tr>
<tr>
<td>5 × 5</td>
<td>213s</td>
<td>47,807,136</td>
</tr>
<tr>
<td>6 × 6</td>
<td>22h 50m</td>
<td>181,718,080</td>
</tr>
<tr>
<td>≥ 19 × 19</td>
<td>≥ 1</td>
<td>≥ 1</td>
</tr>
<tr>
<td>20 × 20</td>
<td>-</td>
<td>0</td>
</tr>
</tbody>
</table>

Number of possible marker grids.