PinTags: A Visual Fiducial Marker System for Logistics

Vishwesh Vhavle

Indraprastha Institute of Information Technology Delhi, India

Abstract. In the realm of postal services and e-commerce, operators increasingly rely on complex automation systems to meet growing demands. Despite the advancements, certain areas still necessitate human intervention. One such challenge is that of Automated Singulation, which involves picking up a parcel from a bulk shipment and placing it onto a conveyor belt, a task that continues to be a bottleneck for automation systems. PinTags introduces a high-capacity visual fiducial marker system designed to address these challenges. By supporting up to 32,768 unique tags, PinTags not only identifies the pincode of the package's destination but also simultaneously estimates the pose and depth to aid tasks like grasping and manipulation for robotic systems during automated singulation. PinTags aims to significantly reduce the need for manual intervention by introducing an easy-to-integrate, low-cost, and scalable perception solution that could enhance the overall efficiency of postal and e-commerce logistics. Our marker generator is available at https://pintag-review.github.io/PinTag/.

Keywords: Visual Fiducial Markers · Robotics · Computer Vision · Logistics Automation

1 Introduction

Visual fiducial markers, also known as fiducial tags or artificial landmarks, are distinctive visual patterns designed for easy detection and identification by computer vision systems. These markers serve as reference points in an environment, enabling various applications in robotics, augmented reality, and computer vision. One of the most significant advantages of visual fiducial markers is their ability to provide precise 6-degree-of-freedom (6-DoF) pose estimation, which includes both position and orientation information [1, 2].

The widespread adoption of visual fiducial markers in robotics research can be attributed to their integration into popular software frameworks such as Robot Operating System [3] and OpenCV [4]. Libraries like ArUco, integrated into OpenCV, have made it easier for researchers and developers to implement marker-based solutions in their projects [5].

Beyond basic pose estimation, visual fiducial markers have found applications in more complex tasks. For instance, in the field of Simultaneous Localization and Mapping (SLAM), TagSLAM utilizes fiducial markers to achieve robust SLAM

performance in challenging environments [6]. In underwater robotics, where traditional vision-based methods often fail due to poor visibility and distortion, AprilTag-based SLAM has been successfully employed for high-precision robot localization [7].

An innovative application of visual fiducial markers is exemplified by NaviLens, a system designed to assist visually impaired individuals [8]. NaviLens markers have been adopted in various public spaces, including subway stations, restaurants, and product packaging, to provide location-based information and guidance.

Drawing inspiration from these diverse applications, we propose PinTags, a high-capacity visual fiducial marker system designed to address the challenges of automated singulation in logistics. Automated singulation refers to the process of separating individual items from a bulk shipment, typically for sorting or further processing [9]. This task remains a significant bottleneck in logistics automation, often requiring human intervention due to the complexity of grasping and manipulating diverse package shapes and sizes.

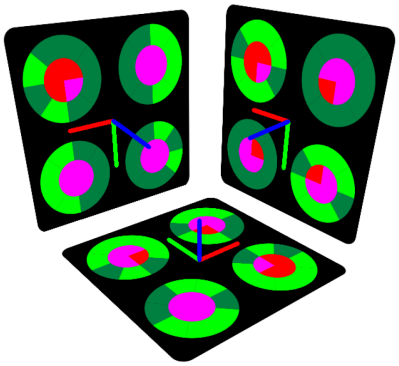


Fig. 1: An example illustrating three detected PinTag markers along with plotted estimated pose information. Red, green, and blue lines indicate the estimated X, Y, and Z axes, respectively.

The logistics industry in India presents a unique challenge for automated singulation systems. With over 19,000 unique postal codes or pincodes [10], there is a need for a high-range visual fiducial marker system that can efficiently encode this vast number of destinations while simultaneously providing pose estimation for robotic manipulation. Existing marker systems, while effective for many applications, often lack the capacity to encode such a large number of unique identifiers while maintaining the ability to estimate pose accurately at high detection speeds under varying lighting conditions.

PinTags addresses these challenges by introducing a novel visual fiducial marker system capable of supporting up to 32,768 unique tags. This high capacity not only allows for the encoding of all Indian pincodes but also provides additional capacity for future expansion or encoding of supplementary information. Moreover, PinTags is designed to simultaneously estimate the pose and depth of packages similar to other popular visual fiducial marker systems [2,11,12], facilitating robotic grasping and manipulation tasks during automated singulation.

In the following sections, we will detail the design of PinTags, its detection pipeline and evaluate its performance against the most widely-adopted markers systems [2, 11].

2 Related Works

Visual fiducial marker systems have evolved significantly over the years, with each new system addressing limitations of its predecessors and introducing novel features. ARToolKitPlus, an extension of the original ARToolKit, introduced improvements in detection speed and accuracy. It supported a wider range of marker types and offered better performance on mobile devices [13]. ARTags, developed as an alternative to ARToolKit, provided enhanced robustness to lighting variations and occlusions [1].

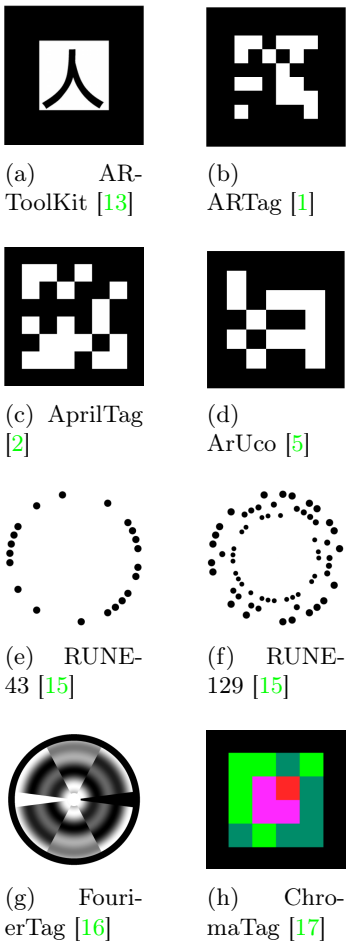
AprilTags represented a significant advancement in fiducial marker technology. These square markers use a 2D binary coding scheme that allows for a large number of unique tags. AprilTags offer improved detection accuracy and robustness compared to earlier systems, particularly in challenging lighting conditions and at longer distances [2]. ArUco markers, part of the OpenCV library [4], offer a balance between simplicity and effectiveness. They use a dictionary-based approach for marker generation and detection, allowing for customizable marker sets. ArUco markers are known for their computational efficiency and good performance across various lighting conditions [5, 11].

AprilTag2 built upon the success of the original AprilTag system with several key improvements to its detection pipeline [12]. These include enhanced edge detection, employing a more sophisticated method for identifying marker boundaries and improving detection in complex environments. AprilTag2 also features refined corner estimation, using a gradient-based approach for more accurate corner localization, crucial for precise pose estimation. AprilTag3 [14] further advances the system by introducing flexible tag layouts, including the 'uramaki' design that allows data bits outside the marker border. It also introduces circular and recursive tag layouts, catering to specific applications. The detection algorithm in AprilTag3 is further optimized resulting in faster detection and higher recall rates than AprilTag2, and it introduces new families of tags like 10x10 Uramaki marker with 48714 markers, however most AprilTags have very poor detection speeds.

RuneTag introduces a circular marker design, offering two main variants: RUNE43 and RUNE129 [15]. RUNE43 provides a relatively small set of 762 unique tags, while RUNE129 significantly expands the range to 19,152 tags. However, RUNE129 faces challenges due to the dense packing of circular points, which can lead to incorrect feature merging by ellipse detectors at low resolutions, effectively reducing the maximum detection distance. RUNE129 markers also perform poorly at steep viewing angles, limiting their usability in certain scenarios.

Fourier Tags represent a departure from traditional binary coding schemes [16]. By encoding information in the frequency domain, Fourier Tags offer enhanced robustness to perspective distortions and partial occlusions. This approach allows for a more continuous degradation of performance under challenging conditions, rather than the abrupt failures often seen in binary-coded systems. Fourier Tags require more complex image processing algorithms, including Fourier transforms, which can significantly increase computational demands. This leads to slower detection times, especially on resource-constrained devices.

ChromaTag introduces color into fiducial marker design, leveraging the LAB color space for improved robustness to lighting and color variations [17]. By using color information, ChromaTag can achieve faster initial detection compared to grayscale-based systems. Operating in the LAB color space allows ChromaTag to maintain consistent performance across various lighting conditions and camera color balances. The use of color ratios rather than absolute color values contributes to ChromaTag's robustness against lighting changes. These innovations make ChromaTag particularly suitable for applications in dynamic lighting environments or when using cameras with varying color characteristics.



(a) ARToolKit [13] (b) ARTag [1] (c) AprilTag [2] (d) ArUco [5] (e) RUNE-43 [15] (f) RUNE-129 [15] (g) FourierTag [16] (h) ChromaTag [17]

Fig. 2: Examples of different visual fiducial marker systems: (a) ARToolKit, (b) ARTag, (c) AprilTag, (d) ArUco, (e) RUNE-43, (f) RUNE-129, (g) Fourier Tag, and (h) ChromaTag.

Most visual fiducial marker systems define a fixed library of tags that can recover encoded data and tag pose. These systems like [2, 5, 17] generally emphasize handling occlusions, improving detection speeds and maintaining robustness under various lighting conditions. The evolution of these systems has seen improvements in the number of unique tags supported [15, 16], however these markers have limited practical usability. There appears to be limited work focused on developing an open-source, high-capacity marker systems that also

maintains competitive detection speeds and robustness to varying viewing angles and lighting conditions.

3 Marker Design

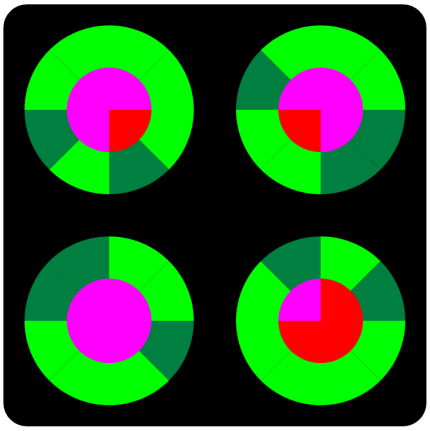


Fig. 3: An example of a PinTag.

The PinTag marker, as illustrated in Figure 3, is designed as a square visual encoding system with four circular elements and each circular element is comprised of a red region and a green region. The red circular region is segmented into 4 sectors, with 4 bits of information using contrasting shades of red which we use to resolve the orientation as explained in Section 4.4, and the green region is segmented into 8 sectors, with 8 bits of information. These bits are color-coded using contrasting shades of green. Both red and green regions together encode a total of 5 parity bits.

The tag represents the encoded information derived from a 6-digit pincode. The encoding process converts valid pincodes to 15-bit hash keys and we compute 5 parity bits using the hash keys. The 15-bit along with 1 parity bit is encoded twice into the tag design symmetrically using the green region. The red region houses 4 parity bits and 12 bits for encoding the orientation. The choice of colors is particularly significant when considered in the context of the LAB color space. The LAB color space, visualized in Figure 4, is designed to approximate human vision. It consists of three channels:

- L: Lightness (0 to 100)
- A: Green-Red component (-128 to +127, from green to red)
- B: Blue-Yellow component (-128 to +127, from blue to yellow)

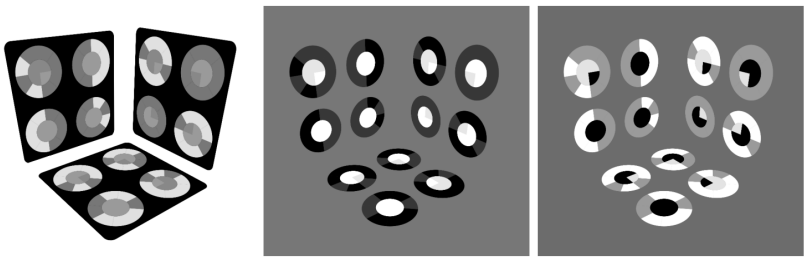


Fig. 4: Visualization of the LAB color space channels: L (left), A (center), and B (right).

Similar to ChromaTag [17], PinTag's color selection makes use of the LAB color space as the green-red contrast maximizes the A channel range which allows to quickly detect the red circles in the A channel and we encode the binary code in the B channel using contrasting B values in green and red regions which easy to detect in the B channel. This color strategy enhances the tag's detectability and decodability across various lighting conditions and camera types as LAB is device-independent, meaning that it defines color regardless of the device used to capture or display it.

4 Detector

The PinTag detector first finds red circles then attempts to look for quads in the vicinity of the detected red circles. This method helps improve detection speeds and reduce false positives. Orientation decoding and payload decoding is trivial once we have the quads and their associated centroids as shown in Sections 4.4 and 4.5. Algorithm 1 summarizes the PinTag detection pipeline.

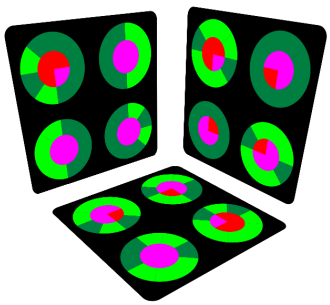


Fig. 5: This input example is intentionally simple and we use it to demonstrate the PinTag detection pipeline.

Algorithm 1 PinTag Detection

Require: Input image I_{RGB}
$I_{LAB} \leftarrow f_{LAB}(I_{LAB})$ ▷ Convert input to LAB color space
$L, A, B \leftarrow I_{LAB}$
$C_r \leftarrow \text{FindRedCircles}(A)$ ▷ Returns centroids of red circles
$Q \leftarrow \text{FindQuads}(L, C_r)$
$G \leftarrow \text{GroupCentroids}(Q, C_r)$ ▷ Group centroids into quads
$i \leftarrow 0$
while $i < \text{length}(G)$ **do**
    $G_i \leftarrow G[i]$
    $T_i \leftarrow \text{GetPerspectiveTransform}(G_i, G_{template})$
    $Red_i, Green_i \leftarrow \text{TransformQuery}(B, T_i^{-1})$
    $O_i \leftarrow \text{DecodeOrientation}(Red_i)$
    $Value_i \leftarrow \text{DecodeGreenSectors}(Green_i, O_i)$
    $FinalValue_i \leftarrow \text{ParityCheck}(Value_i)$
    $R_i, t_i \leftarrow \text{EstimatePose}(G_i, T_i)$
    results $\oplus$ $(FinalValue_i, R_i, t_i)$
    $i \leftarrow i + 1$
end while
return results

4.1 Red Circle Detection

Red circles are detected using thresholding on the A-channel of the LAB color space. Contours are found and filtered based on circularity and size. Circularity is measured using the formula Circularity $= \frac{4\pi \cdot \text{Area}}{\text{Perimeter}^2}$, where a value closer to 1 indicates a more perfect circle. Algorithm 2 outlines the red circle detection process.

Algorithm 2 Red Circle Detection

Require: Input image I_{RGB}
$I_{LAB} \leftarrow f_{LAB}(I_{LAB})$ ▷ Convert input to LAB color space
$L, A, B \leftarrow I_{LAB}$
$M_r \leftarrow \mathbf{1}(A > \tau_r)$ ▷ Binary thresholding, we use $\tau_r = 150$ for our experiments
$C \leftarrow \text{FindContours}(M_r)$
$C_{large} \leftarrow \{c \in C : \text{Area}(c) > \tau_a\}$ ▷ Filter small contours
$C_{sorted} \leftarrow \text{Sort}(C_{large}, \text{key} = \text{area})$
$C_{filtered} \leftarrow \{c \in C_{sorted} : \text{IsRound}(c)\}$
$Centroids \leftarrow \{\text{Centroid}(c) : c \in C_{filtered}\}$ ▷ Extract centroids
return Centroids

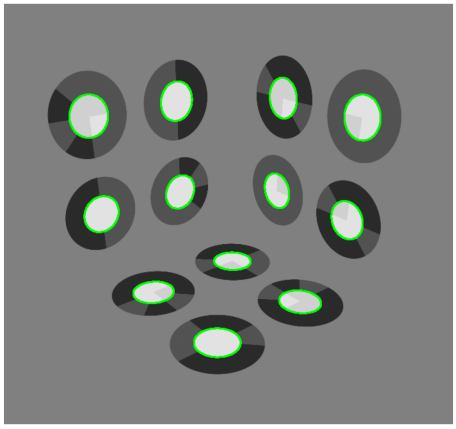


Fig. 6: Detected red circles.

4.2 Quad Detection

Our quad detection procedure is loosely based on the one used by ArUco markers [11], however we opt for Otsu thresholding [18] instead of the Adaptive thresholding used by ArUco makers as using such a global threshold leads to fewer artifacts in the processed image. Further, we use our centroids create a distance-based mask to reduce the region in which quads needs to be detected in. Once the quads are detected, centroids are then grouped into quads.

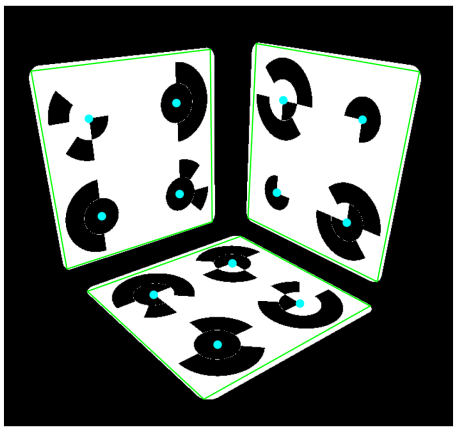


Fig. 7: Detected Quads bounded by green and grouped centroids in blue.

Algorithm 3 Quad Detection

Require: Input image I_{RGB}
1: $I_{LAB} \leftarrow f_{LAB}(I_{LAB}), centroids$
2: $L, A, B \leftarrow I_{LAB}$
3: $M_o \leftarrow \text{OTSU}(L)$ ▷ Otsu thresholding
4: $M_c \leftarrow \textsc{CreateDistanceFilterMask}(M_o, centroids)$ ▷ Create M_c using distance filter mask of centroids
5: $quads \leftarrow \textsc{FindSquares}(M_c)$
6: $G \leftarrow \textsc{GroupCentroids}(centroids, quads)$ ▷ Group centroids into quads
7: **return** G

4.3 Perspective Correction

Perspective correction is performed using the Direct Linear Transform (DLT) method [19]. Given a set of 4 corresponding points in the original image (x_i, y_i) and the destination image (X_i, Y_i), we solve for the homography matrix H:

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \sim H \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix} \tag{1}$$

Where H is a 3x3 matrix:

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \tag{2}$$

The DLT algorithm solves for H by setting up a system of linear equations and using SVD to find the solution.

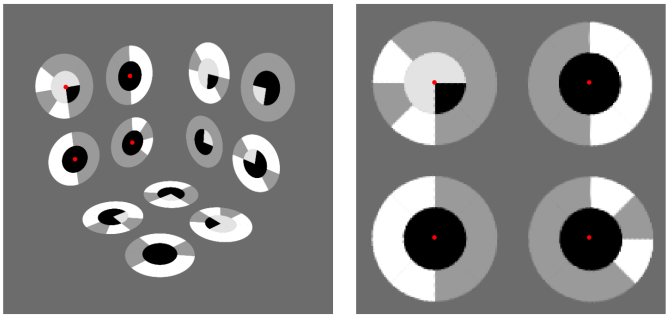


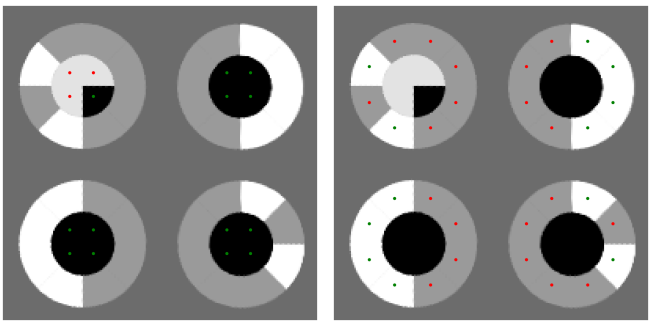
Fig. 8: Perspective transform estimation illustrated for the first marker

4.4 Orientation and Payload Decoding

We use the inverse of estimated perspective transformation H^{-1} to transform and query template query locations in the B-channel and retrieve the pixel values. We then estimate thresholds for the contrasting red regions and green regions.

Orientation Decoding Orientation is decoded by sampling four points around the center in each inner circle and comparing their values to the determined red threshold. We then estimate orientation of the tag. Figure 9(a) shows decoded 0 bits in red and 1 bits green. The outer sectors of three of the circles is set to 1 and outer sectors for the fourth circle is set to 0. The inner sectors store the parity bits.

Payload Decoding The payload is decoded by sampling 8 points in the outer ring and comparing their values to the determined green threshold. We use the estimated orientation to determine the order for parsing. Figure 9(b) shows a blue line to show where parsing begins, decoded 0 bits in red and 1 bits green.



(a) Orientation decoding (b) Payload decoding

Fig. 9: Decoding illustrated for the first marker

5 Results

To evaluate and benchmark PinTag against other widely adopted visual fiducial markers like ArUco marker and AprilTags, we create dataset by rasterizing views of the markers in corresponding poses in simulation as shown in Figure 10. We evaluate against similarly sized ArUco 5x5-1000 and AprilTag 36h10, due to their widespread use and state-of-the-art detectability even though these families of tags offer significantly lower range. Naturally, as we increase the target distance,

performance for all markers decrease gradually, however in our experiments, all three markers have 100% detection upto 5m which is more than sufficient for the intended use cases for PinTags. Figure 11 shows how the accuracy decreases as we shift ϕ.

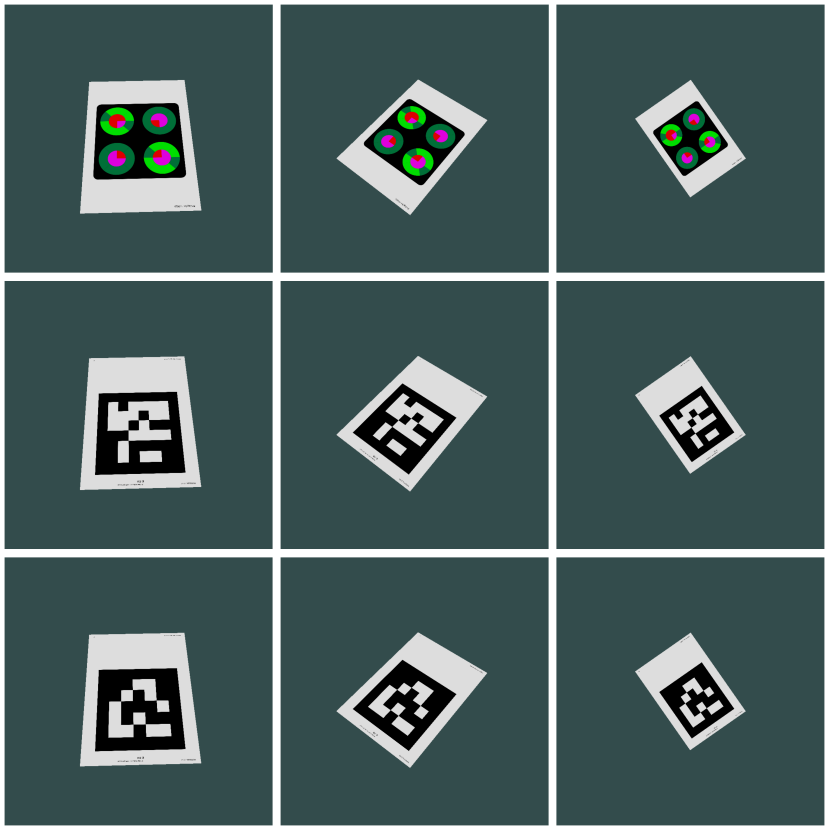


Fig. 10: Corresponding views of PinTags(top), AprilTags(middle), and ArUco markers(bottom)

We can see that while PinTags fall short of the performance when compared to AprilTags and ArUco markers in detectabilty while varying ϕ, it is still quite competitive. Table 1 summarizes the detection times for all three markers and we can see that PinTag has far better detection speed on average than AprilTag. All experiments were conducted on an Intel® Core™ i7-10750H CPU at 2.60GHz

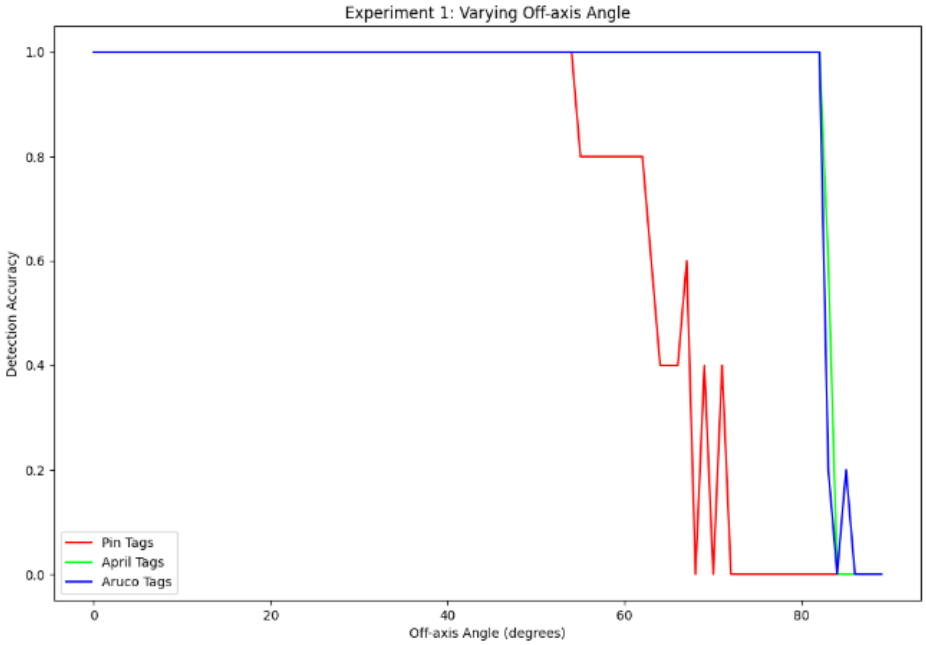


Fig. 11: Detection Accuracy along varying Off-axis Angles

Tag Type	Minimum(s)	Average(s)	Maximum(s)
Pin Tags	0.0106	0.0217	0.0328
April Tags	0.2788	0.3061	0.3633
ArUco Tags	**0.0050**	**0.0075**	**0.0107**

Table 1: Summary of Detection Times for Different Tag Types

6 Conclusions

PinTags represents a novel visual fiducial marker technology supports up to 32,768 unique tags, PinTags offers sufficient capacity to encode all 19,000+ Indian pincodes while providing room for future expansion and additional information encoding. Future work could explore handling partial occlusions, improvements to the data encoding, and more rigorous evaluation in real-world scenarios.

In conclusion, our system builds upon the strengths of existing marker technologies while introducing innovations to meet the specific demands of high-capacity encoding and robust pose estimation with fast detection speeds. PinTags combines the benefits of systems like AprilTags, ArUco and ChromaTag markers with novel design elements to achieve both high capacity and reliable performance under various conditions.

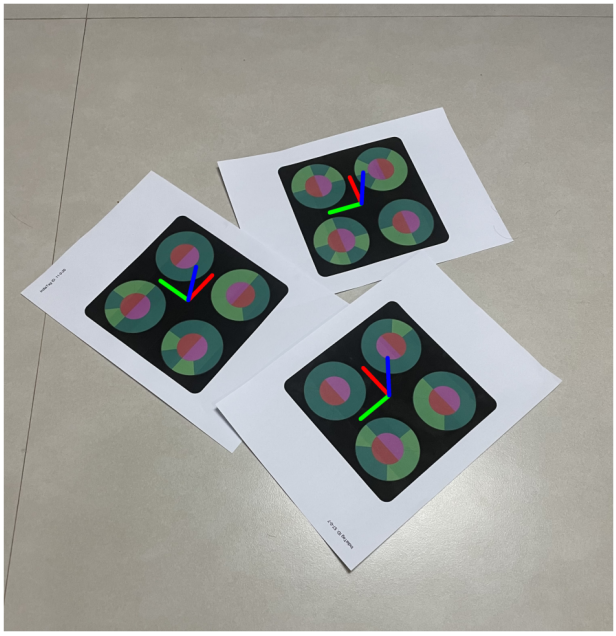


Fig. 12: Detection of PinTags in real world scene.

References

1. M. Fiala, "Artag, a fiducial marker system using digital techniques," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 590–596.
2. E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 3400–3407.
3. M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," 2009, iCRA Workshop on Open Source Software [Online; accessed 07-August-2024]. [Online]. Available: https://www.ros.org
4. G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.
5. S. Garrido-Jurado, R. Muñoz-Salinas, F. Madrid-Cuevas, and M. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014.
6. A. Dehann, D. Fourie, J. Claassens, and J. Leonard, "Tagslam: Robust slam with fiducial markers," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 5052–5057.
7. E. Westman and M. Kaess, "Underwater apriltag slam and calibration for high precision robot localization," *Carnegie Mellon University*, 2018.
8. NaviLens, "Navilens: A new way to perceive the world," https://www.navilens.com/, 2021, [Online; accessed 07-August-2024].

9. C. Eppner, S. Höfer, R. Jonschkowski, R. Martín-Martín, A. Sieverling, V. Wall, and O. Brock, "Lessons from the amazon picking challenge: Four aspects of building robotic systems," in *Robotics: Science and Systems*, 2016.
10. Government of India, "All india pincode directory," https://www.data.gov.in/resource/all-india-pincode-directory-till-last-month, 2023, [Online; accessed 07-August-2024].
11. R. Munoz-Salinas and S. Garrido-Jurado, "Aruco library," http://sourceforge.net/projects/aruco/, 2013, [Online; accessed 07-August-2024].
12. J. Wang and E. Olson, "Apriltag 2: Efficient and robust fiducial detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4193–4198.
13. D. Wagner and D. Schmalstieg, "Artoolkitplus for pose tracking on mobile devices," in *Proceedings of 12th Computer Vision Winter Workshop*, 2007.
14. M. Krogius, A. Haggenmiller, and E. Olson, "Flexible Layouts for Fiducial Tags," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nov. 2019, pp. 1898–1903, iSSN: 2153-0866. [Online]. Available: https://ieeexplore.ieee.org/document/8967787
15. F. Bergamasco, A. Albarelli, L. Cosmo, E. Rodolà, and A. Torsello, "An accurate and robust artificial marker based on cyclic codes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2016.
16. A. Xu and G. Dudek, "Fourier tag: a smoothly degradable fiducial marker system with configurable payload capacity," in *Computer and Robot Vision (CRV), 2011 Canadian Conference on*. IEEE, 2011, pp. 40–47.
17. J. DeGol, T. Bretl, and D. Hoiem, "ChromaTag: A Colored Marker and Fast Detection Algorithm," Aug. 2017, arXiv:1708.02982 [cs]. [Online]. Available: http://arxiv.org/abs/1708.02982
18. N. Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, pp. 23–27, 1975.
19. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge, UK: Cambridge University Press, 2004.