

An Efficient Visual Fiducial Localisation System

Peter Lightbody
Lincoln Centre for
Autonomous Systems
University of Lincoln
United Kingdom
plightbody@lincoln.ac.uk

Tomáš Krajník
Artificial Intelligence Center
Czech Technical University
Czechia
tomas.krajnik@fel.cvut.cz

Marc Hanheide
Lincoln Centre for
Autonomous Systems
University of Lincoln
United Kingdom
mhanheide@lincoln.ac.uk

ABSTRACT

With use cases that range from external localisation of single robots or robotic swarms to self-localisation in marker-augmented environments and simplifying perception by tagging objects in a robot's surrounding, fiducial markers have a wide field of application in the robotic world. We propose a new family of circular markers which allow for both computationally efficient detection, tracking and identification and full 6D position estimation. At the core of the proposed approach lies the separation of the detection and identification steps, with the former using computationally efficient circular marker detection and the latter utilising an open-ended 'necklace encoding', allowing scalability to a large number of individual markers. While the proposed algorithm achieves similar accuracy to other state-of-the-art methods, its experimental evaluation in realistic conditions demonstrates that it can detect markers from larger distances while being up to two orders of magnitude faster than other state-of-the-art fiducial marker detection methods. In addition, the entire system is available as an open-source package at <https://github.com/LCAS/whycon>.

CCS Concepts

• **Computing methodologies** → *Object detection; Tracking; Vision for robotics;*

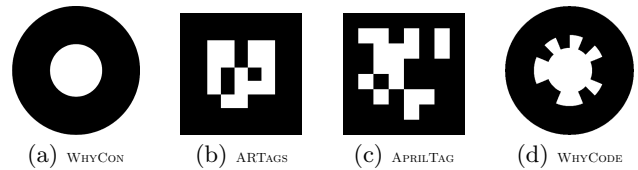
Keywords

Fiducial Markers; Swarm Robotics; Necklace Code; Visual Tracking

1. INTRODUCTION

Although initially intended for Augmented Reality (AR) applications, fiducial-based visual localisation systems are broadly utilised in a number of areas throughout the field of robotics where a robust and efficient full pose vision-based estimation is required. Typical applications of such marker-based systems include swarm and bio-inspired robotics [2, 3, 11], which requires reliable localisation of a large number of robots from an external camera (see also Figure 1(e)), visual-

Copyright is held by the authors. This work is based on an earlier work: SAC'17 Proceedings of the 2017 ACM Symposium on Applied Computing, Copyright 2017 ACM 978-1-4503-4486-9. <http://dx.doi.org/10.1145/3019612.3019709>



(e) Swarm of robots tagged with WhyCode markers

Figure 1: Four types of fiducial markers: the state-of-the-art WhyCon, ARTags, AprilTag and the proposed WhyCode and a robotic swarm tagged with the WhyCode markers.

serving that requires highly precise robot motion [21, 26], and semantic scene understanding [7], in which the scene objects are tagged with the fiducial markers to mitigate the limitations of general vision-based object recognition.

Both within these applications and more generally, visual fiducial marker detection and tracking systems must ideally fulfil to a high standard, the following requirements:

- **Robustness:** Markers must be robustly detectable in adverse conditions such as when moving rapidly, when at a considerable distance or when under varying lighting conditions.
- **Distinguishability:** For fiducial markers to be used for applications in which a single marker is not sufficient, either because several robots need to be tracked in parallel or several features in the environment need

to be identified simultaneously, it needs to robustly identifiable and distinguishable to the vision system. However, the number of markers required in these applications often varies considerably across application domains. The tracking method must, therefore, be able to scale accordingly to the requirements imposed by the specific application or scenario.

- **Economic Feasibility:** To ensure the system is universally accessible, it should, ideally, utilise markers which are cheap and easily producible in large quantities, such as being printable on paper. This has the added benefit of making the system not only cheap to set up, but also cheap to use in conditions which might otherwise deteriorate the markers such as use with drones or in adverse weather conditions. The system should also utilise standard, off-the-shelf sensor(s) such as RGB and grey-scale cameras. All of this, together with freely-available, open-source software, makes them customisable, and thus, minimises the cost for developers and researchers alike.
- **Precision:** Fiducial markers’ fundamental purpose is to provide a precise position of a marked object within an observed scene. This can come in the form of a 2D location in an image or a 3D position in the scene. Apart from identifying a 3D position, orientation information is often also desirable. Thus, most markers often have properties that allow the estimation of their full 6 degrees of freedom (DoF) pose.

In this paper, we propose a novel tracking system that can generate suitable markers which can be easily printed on paper, along with an integrated software component that addresses the above requirements to a very high standard. This is achieved by extending an open-source detection system for circular markers called WHYCON [12] by adding a novel encoding based on the concept of *Binary Necklaces* [20], which we shall refer to as WHYCODE. Necklaces are a mathematical concept of combinatorics providing a *generator* for rotation invariant, uniquely identifiable patterns that can scale to a theoretically infinite number of individual markers, similar to the one shown in Fig. 1(d). The resulting markers are robustly and efficiently detectable in the environment and also allowing for discrimination between individual markers using the Necklace coding. With our extension of the original system we now present a 6-DoF fiducial marker system. The performance of the proposed system is demonstrated through a range of experiments which compare the pre-existing WHYCON system against the new WHYCODE method, in addition to the frequently used ARTAGS and APRILTAG fiducial marker detection systems.

2. RELATED WORK

In response to the need for vision-based markers within robotics, discussed above, several marker-based tracking and identification methods have been developed. These can be categorised into three groups: active markers, passive reflective markers and passive markers, depending on the type of markers that each system uses.

In vision based systems, active markers contain a source of light that can be tracked – this is often achieved using an infra-red or ultraviolet LEDs, which are easy-to-detect by cameras, but unobtrusive to humans. These systems also achieve very low false positive rates, because the active markers can emit unique identification codes, which makes them almost impossible to confuse with other objects. Some active marker systems are also capable of estimating the orientation of a marker, but this is normally achieved by combining the positions of a number of markers to calculate the orientation, rather than retrieving the orientation of a single marker. These systems, however, do offer sub-millimetre precision for a marker’s location and their multicamera setups offer extremely accurate tracking information [22].

Alternatively, passive reflective markers are a widely used approach within the field, with the most common example being the commercial motion capture system ViCon [24], which combines high-resolution and high speed cameras that utilise strong infra-red emitters. Systems like this also enable tracking with sub-millimetre precision but unlike active marker systems, their fiducials do not contain electronic components. They are instead often made from, or coated with, materials that are highly IR reflective, allowing the IR cameras to easily pick out the markers from a scene even from high distances. This has the benefit of making the markers cheaper and easier to deploy than active ones, however, passive systems are problematic to deploy in areas under direct sunlight. Although both active and passive reflective based systems offer a solid ground truth, these approaches remain very costly and are therefore not always an appropriate solution.

These issues, however, have motivated the creation of a variety of alternative low-cost tracking systems which focus on the final category of fiducial markers: passive vision-based tracking. With many of these newer methods utilising simple planar patterns, it is not only the cost that is significantly lowered, but also the difficulty of use and set up time.

One of the most well-known passive markers is the QR Code. This marker is predominantly designed to store information, such as text and URLs, and consists of a two-dimensional matrix barcode which encodes data in a pattern of black and white squares. In-built error correction codes allow the information to be correctly read, even if the marker is partly damaged, although these characteristics do restrict the range and angles from which the codes can be read. Consequently, although there is the potential to use such markers as part of a larger tracking system, their design makes them less suitable for tracking than both the methods discussed below and the proposed method.

Examples of passive markers which are more focused on the tracking element are the augmented-reality markers. These, although conceptually similar to the QR code mentioned above, are designed to encode far smaller data payloads and often use the ARTag [8] and ARToolKit+ [25] software libraries.

The current ARTAGS developed from these software libraries utilises a square box fiducial marker which encodes information through the use of a large 2D black and white bar code. The real time performance of the system, coupled with its

accuracy and robust nature, make it an ideal candidate for a comparison to the proposed system.

Another augmented-reality alternative that will also be compared is the APRILTAG [18] system. Using the square marker design with a 2D bar code, the APRILTAG also stems from a lexicographic coding system [23] and is therefore able to be detected at both short and long range. Computational simplicity is, however, sacrificed.

The success of square markers within this field is evident, however, the use of circular markers is quickly becoming a regular occurrence in many applications. This is largely due to the need to counter the expensive operation required to correct the shifting of the centroid of a square marker under perspective transformation. Many systems, including the SyRoTek e-learning platform [14], which uses ring-shaped patterns with binary tags and [26], a planar pattern which consists of the letter 'H' surrounded by a ring, utilise circular markers due to the less expensive centroid operation. In the latter system, the pattern is first detected using adaptive thresholding and is later processed for connected component labelling. To establish whether the marker has been correctly tracked, its geometric properties are tested and the false matches are discarded. A Canny edge detector and ellipse fitting method are then applied to the positive matches.

Another system built upon the same methodology as the ARToolKit and ARTags is ArUco [10], which boasts a robust ID system with an error correction technique that can handle up to 1024 individual codes. The detection process within the ArUco system combines contour extraction and code identification with the aforementioned adaptive thresholding step and thus can determine the extrinsic parameters of the marker using the intrinsic camera parameters.

Finally, a system relatively similar to the proposed markers, that comprises a number of concentric circles broken into several angular regions and coloured either black or white, is the TRIP localisation system [6], which is able to distinguish between 39 patterns. Similarly to the ArUco system mentioned above, the TRIP localisation system also appropriates an adaptive thresholding method, with the system as a whole extracting the edges of the markers and processing the edges which correspond to the circular border of the ring patterns. The main disadvantage accompanying this system is the computational cost, as the adaptive thresholding and ellipse fitting are computationally expensive. On the other hand, the system's ability to achieve a precision of between 1% and 3% of relative error may be said to counteract this disadvantage.

The aforementioned methods are widely considered to be the state-of-the-art methods currently existing within the field. Despite this, the real-world performance and low computational cost of the method proposed here makes it potentially superior in several application domains. The ability to expand the recognisable patterns by incorporating a scalable identification code makes the proposed method preferable in cases when one needs a computationally efficient method to detect, localise and identify a larger number of fiducial markers.

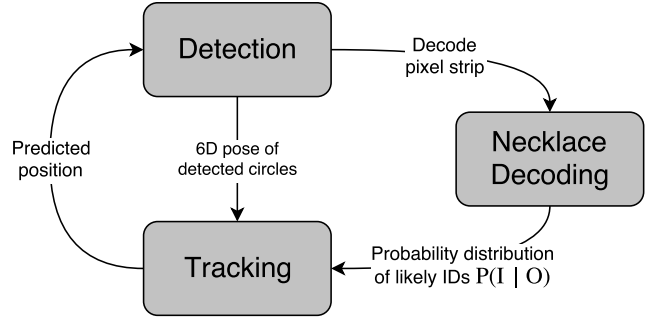


Figure 2: Overview of the approach.

3. CONCEPT

The proposed system builds upon the strengths of the original WHYCON system, with the general conceptual design indicated in Fig. 2. Oncoming images are analysed following the original WHYCON approach, searching for circular patterns as detailed in Sec. 4. One of the main advantages of the WHYCON system is its ability to start searching for a marker from any position in the image without any performance penalty. Thus, the use of the tracking information to predict the next position of the marker in the image results in a significant performance boost – in case when the prediction is correct, the method processes only those pixels that belong to the marker. Apart from the computational performance boost, the tracking also allows the system to employ Bayesian methods, which enhance the robustness of the system to marker misidentification by taking into account the markers' identification history. The robustness of marker identification can be further improved by employing self-corrective codes on top of the necklace encoding. The motion prediction, that speeds up the image processing, self-corrective codes that detect misidentified markers, and a Bayesian state estimation that improves the identification robustness, are the main improvements to the system originally described in [16]. With these improvements the system presented outperforms its earlier versions [16, 13] in terms computational efficiency, orientation estimation accuracy and identification reliability.

4. DETECTION & LOCALISATION

The WHYCON algorithm was originally intended to localise a large number of concentric black and white circles, of known diameter, in a computationally efficient way. The article [13] shows that the method achieves the same precision as state-of-the-art black and white pattern detectors while being faster by an order of magnitude. To detect the circular pattern, the algorithm searches an image using a combination of flood-fill technique and on-demand thresholding, while gathering statistical information about the patterns on the fly. The statistics gathered allow rapidly identified false candidates to be rejected early in the processing pipeline. One of the key enabling factors in the computational simplicity of the proposed system is that the pattern search can be initiated from any position within the image. By combined this with efficient tracking, the algorithm is typically able to process only the pixels that are occupied by the pat-

terns, leading to significant performance boosts.

For the initial pattern detection phases, the image is searched for a continuous segment of black pixels, which are classified by an adaptive thresholding method that ensures a good level of robustness to adverse lighting conditions. Once a continuous segment of black pixels is found by the flood-fill method, a simple circularity test can be performed.

A pattern consisting of s pixels, with bounding box dimensions b_u, b_v and inner and outer diameters d_i, d_o is considered circular if its ‘roundness’ ρ_{out} is smaller than a predefined value ρ_{max} , i.e.

$$\rho_{max} > |\rho_{out}| = \left| \frac{\pi b_u b_v}{4s} \frac{d_o^2 - d_i^2}{d_o^2} - 1 \right|. \quad (1)$$

Once the black segment passes the circularity test, a new flood-fill search is initiated to locate the inner white segment. Once the inner segment is found, the algorithm then compares the position of the inner and outer segments’ centre points to verify that the segments are concentric. From here, the algorithm then calculates the ratio of inner and outer segments’ pixels to verify that this ratio conforms to the known ratio of the black and white segments’ areas.

After passing these tests, the positions of the segments’ pixels u_i, v_i that were stored during the flood-fill search are used to calculate the pattern’s centre u, v and covariance matrix \mathbf{C} as follows:

$$\mathbf{C} = \frac{1}{s} \sum_{i=0}^{s-1} \begin{pmatrix} u_i u_i & u_i v_i \\ u_i v_i & v_i v_i \end{pmatrix} - \begin{pmatrix} uu & uv \\ uv & vv \end{pmatrix}. \quad (2)$$

Note that u_i, v_i are integers, and the computationally most expensive part of Equation 2 is calculated using integer arithmetic. The u_i, v_i and \mathbf{C} actually represent an elliptical projection of the pattern in the image.

Then, the eigenvalues λ_0, λ_1 and eigenvectors v_0, v_1 of the covariance matrix \mathbf{C} are calculated and used to determine ellipse semiaxes e_0, e_1 as follows:

$$\begin{aligned} e_0 &= 2\lambda_0^{\frac{1}{2}} v_0, \\ e_1 &= 2\lambda_1^{\frac{1}{2}} v_1. \end{aligned} \quad (3)$$

As the length of the ellipse semiaxes is now known, a final segment test can be performed, which verifies if the number of pixels s corresponds to the area of the ellipse:

$$\xi > |\pi e_0 e_1 / s - 1|. \quad (4)$$

The constant ξ represents a tolerance value much lower than ρ_{max} , because the ellipse dimensions e_0, e_1 are obtained from the covariance matrix with sub-pixel precision. If the detected segments satisfy Equation 3, they are assumed to represent the pattern. The obtained eigenvalues and eigenvectors are then used to calculate the spatial position of the pattern.

To obtain the relative distance of the pattern, the pixel coordinates of the ellipse (co-)vertices are calculated and transformed into canonical camera coordinates using the intrinsic camera parameters that were obtained through standard camera calibration procedure. The transformed coordinates of the (co-)vertices are used to calculate the centre and axes

of the ellipse in the canonical camera form. The vertices are used to calculate a conic \mathbf{Q} such that all the ellipse points u', v' satisfy

$$\begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix}^T \mathbf{Q} \begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} = 0. \quad (5)$$

Then, we calculate the eigenvalues $\lambda_0, \lambda_1, \lambda_2$ and eigenvectors q_0, q_1, q_2 of the conic \mathbf{Q} and use them to obtain the spatial position of the pattern by the method presented in [26]:

$$\mathbf{x} = \frac{d_o}{\sqrt{-\lambda_0 \lambda_2}} \left(s_1 q_0 \lambda_2 \sqrt{\frac{\lambda_0 - \lambda_1}{\lambda_0 - \lambda_2}} + s_2 q_2 \lambda_0 \sqrt{\frac{\lambda_1 - \lambda_2}{\lambda_0 - \lambda_2}} \right), \quad (6)$$

where d_o is the circular pattern diameter.

In this work, we also implement a calculation of the patterns orientation. At first, we calculate the normal \mathbf{t} by

$$\mathbf{t} = \left(s_1 q_0 \sqrt{\frac{\lambda_0 - \lambda_1}{\lambda_0 - \lambda_2}} + s_2 q_2 \sqrt{\frac{\lambda_1 - \lambda_2}{\lambda_0 - \lambda_2}} \right). \quad (7)$$

Note that the constants s_1 and s_2 are undetermined signs that have to be selected so that the \mathbf{n} points towards the camera and x is in front of it. In other words, s_1 and s_2 are chosen so that the inequalities:

$$\begin{aligned} \mathbf{n}(0, 0, 1)^T &< 0 \\ \mathbf{x}(0, 0, 1)^T &> 0 \end{aligned} \quad (8)$$

are satisfied. While the roll and pitch of the pattern can be expressed from the normal \mathbf{n} , the yaw of the original circular marker can not be determined. However, the yaw can be calculated in the subsequent step, which uses the Necklace encoding for the pattern identification.

4.1 Motion Prediction

As mentioned before, the flood-fill procedure, which constitutes the core of the segmentation, can be initiated from any point in the image being analysed.

If initiated near, or ideally inside of, the searched pattern, the method will process only the pixels of the pattern itself, which significantly reduces image processing time. Thus, the method’s computational efficiency relies on its ability to reuse the patterns past movement information to correctly predict its position in the image currently being processed. In the earlier versions of the system, the search simply started at the pattern’s position in the last analysed image. While this is sufficient for slowly moving targets, (e.g. in swarm robotics experiments [3]), rapidly moving robots require the system to take into account their velocity in order to maximise the tracking performance [19]. Thus, the motion prediction presented in this work uses several past detections of the pattern to estimate its velocity in image coordinates. The last estimated velocity along with the last detected position are then used to predict the position of the pattern in the currently processed image. The experiments described in 6.7 indicates when tracking rapidly moving targets, the average image processing time can be reduced by more than $\sim 20\%$.

4.2 Automatic Parameter Tuning

Apart from tracking, the swiftness of the system depends on its ability to quickly reject false pattern candidates based on on-the-fly-calculated statistics. However, the decision to reject a segment in each statistical test (e.g. (1) or (4)) requires setting a certain tolerance range. This, in turn, rises the need to set these tolerance values depending on a particular application and hardware used. For example, rolling cameras on a quadrotor suffer from a specific ‘jitter’ or ‘trembling’ noise, caused by vibrations induced by the drone’s motors [1], which makes the segments appear deformed and non-circular. This requires ξ value in Eq. 4 to be relaxed in order to work. Another tolerance value that is affected by hardware used is the expected ratio of black and white pixels in the segment – this is subject to non-linear sensitivity and chromatic aberration of the camera which also depend on the current lighting conditions. To cope with the dependence of these tolerances on the hardware used and current illumination conditions, we employed a simple scheme, which slowly adapts these tolerances to the values that the system experiences during its operation. In particular, if the detection of a given pattern is successful, the expected black-to-white pixel ratio b_{exp} is updated as follows: $b_{exp} \rightarrow 0.99 b_{exp} + 0.01 b_{real}$, where b_{real} is the value calculated by the segmentation method. The other parameters, ρ , ξ etc. are updated analogously.

5. MARKER IDENTIFICATION

Building upon the good detection performance of the WHYCON system, and adhering to the requirement outlined in the introduction, our development of a new marker system focused on creating a marker which is compatible with the circular features of WHYCON, but also capable of providing a scalable encoding system to uniquely identify each marker. The proposed encoding chosen for the WHYCODE marker was originally identified within the combinatorics field of mathematics, and currently used widely in the fields of combinatorial chemistry [4] and computational biology [5]. These sequence patterns known as Necklaces are “lexicographically the smallest element in an equivalence class of strings under string rotation” [20].

Despite it currently not being used in the field of robotics, this encoding was a highly suitable option for the proposed system due to its rotational invariant nature. By bit-rotating the detected sequence until its lowest binary value is reached, the system is able to identify a starting point regardless of the position from which the code was originally read from. This technique of altering the detected code without confusing the IDs is the core concept being identifying the markers’ yaw rotation. Taking the number of rotations required to reach the lowest binary value, we are able to identify how far rotated the marker is, from a ‘zero point’ of rotation, circumventing the issue of identifying a starting point on a circular marker. This rotation can then be transformed into 3D space to calculate the marker’s yaw rotation, making the markers position detectable in a full 6th DoF. As the ID is encoded by bit-rotating each number to their lowest binary value, both the ID calculation and subsequent yaw rotation can both be pre-calculated and stored to minimise computational costs, thus improve the performance of the

system. However, for this to work reliably all codes which have rotational symmetry, must also be removed from the encoding system, as they allow for the lowest binary value to be reached from multiple start locations, which would result in ambiguity when establishing the markers’ yaw. To see an example of a marker with ambiguous yaw, see the leftmost quad-copter on Figure 1(e).

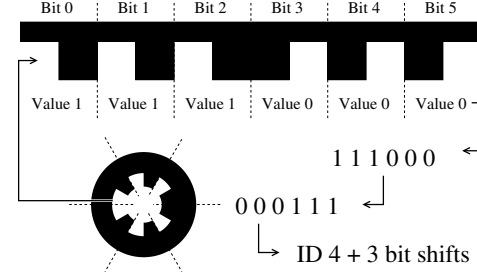


Figure 3: An example of how the Manchester Encoding is used with the Necklace System: The inner circle of the WhyCode marker encodes a binary string which is bit-rotated to match a Necklace code. Apart from identification, the number of bit-rotations allows us to identify the marker’s rotation.

To create a system which reliably identifies the markers and preserves backward compatibility with the WHYCON marker, we encoded the Necklace-based ID into the inner circle of the tags and used Manchester Encoding [9]. Thus, each individual bit of the Necklace code is encoded by two consecutive segments of opposite colour, as demonstrated in Figure 3. Although the use of Manchester Encoding halves the number of segments available on the marker, it allows us to calculate an identification confidence rating based on the expected number of pixels in each segment of the Necklace code.

In theory, the Necklace Encoding supports higher than binary bases, and it would be possible to encode the marker IDs in grey-scale values along the inner rim of the circle. However, preliminary tests showed that the edge-based Manchester Encoding is more suitable due to its robustness. This has the benefit of making the system more robust, especially when subject to various lighting condition, but does have the negative effect of only allowing binary-code sequences when encoding IDs. As a result, this restricts the encoding system and limits the number of potential IDs to:

$$N = \frac{1}{n} \sum_{d=1}^n \varphi(d) 2^{n/d}, \quad (9)$$

where $\varphi()$ is totient function [15] and n is the Necklace code length in bits. The Equation 9 is further illustrated in Table 1 which shows the number of combinations valid for the proposed marker, given that the Necklace code consists of a sequence of n bits:

5.1 Identification Uncertainty Handling

Decoding the ID actually requires that one establishes correctly the falling and rising edges of the Manchester code that is encoded in the inner circle of the pattern, see 3.

Table 1: Necklace code length in bits and corresponding number of unique marker identities

Code length [bit]	4	6	8	10	12	14	16
Unique IDs	3	9	30	99	335	979	2623

Thus, the intensity of these edges in the image actually corresponds to the quality of the decoding ring image. If the pattern is close to the camera, one can expect that these edges are prominent. However, if the pattern is far away, or if the image is subject to motion blur, the edges will be weaker and the ID might not be decoded correctly.

Although the separation of encoding and tracking allows the marker to be tracked at distances far beyond the point at which it can be identified, it is important that the system provides consistent ID information even if the individual ID decoding is noisy. In order to handle noisy ID readings, we maintain a probabilistic distribution of all possible IDs over all visible markers. This distribution is updated using Bayes whenever a detected marker ID is decoded. In other words, each detected marker is associated with n numbers, representing the probabilities $p(id = 1 \dots n)$ that the marker has an ID $1..n$. Whenever the marker's ID is decoded, each of these numbers is updated by a Bayesian rule

$$p(id|o) = \frac{p(o|id)}{p(o)}p(id), \quad (10)$$

where $p(id)$ represents the prior probability of the marker having a given id and $p(o|id)$ corresponds to the probability of decoding the pattern incorrectly. Since $p(o|id)$ is determined by the quality of the marker's decoding ring image, we quantify the quality of the decoding ring as the sum of gradients along it – this corresponds to the quality of the edges that constitute the Manchester encoding. Thus, we assume that the probability of the correct decoding of $p(o == id)$ equals $f(s)$, where s is the sum of the gradients along the image of the decoding ring and $f(\cdot)$ is a sigmoid-shaped function (modelled by an $\arctan(\cdot)$ in our case). Our experiments have shown that $f(s)$ captures many of the factors which affect the marker's identification correctness, such as marker size, detection distance, motion blur, extreme viewing angles and low-lighting conditions. Thus, in cases, where the edges of the decoding ring are weak, the confidence $f(s)$ is low and the id of the marker is affected more by its previous state than by the current observation. However, if a marker is particularly visible, $f(s)$ is high and the new, high quality measurement is taken into account. Note that this calculation is computationally inexpensive and as such does not affect the computational complexity of the system.

5.2 Hamming Code

Another possibility to improve the quality of the pattern identification is the adoption of self-corrective encoding on top of the necklace code. The first logical step was to increase the number of code bits n and ensure that a minimum Hamming distance w is kept between any two codes used. In theory, this would allow for the correction of $(w-1/2)$ bit errors, thus giving the system the ability to identify $(w-1)$ errors in bit order. This would consequently increase the

robustness of the system because if a bit or two flips, it is regarded as invalid and is rounded to the closest ID. Because the proposed coding system is rotation invariant, however, it also needs to take into account the Hamming distance between every cyclic permutation of the necklace code. This can severely limit the number of possible IDs, see Table 2. Moreover, more bits means smaller elements of the decoding

Table 2: Necklace code length in bits and corresponding number of unique marker identities

Hamming distance	Code length [bit]						
	4	6	8	10	12	14	16
1 bit	3	9	30	99	335	979	2623
2 bit	2	5	16	51	170	580	1759
3 bit	1	2	3	7	16	48	125
4 bit	1	1	2	5	10	24	69

ring, which increases the chance of errors when decoding the ID from the image. Thus, there is a trade-off – higher Hamming distance improves a change of detecting or correcting an error, but it also increases a chance of introducing one.

To determine which Hamming distance to use for pattern identification, we recorded 4 sequences, where UAVs carried tags with Hamming distances of 1, 2, 3 and 4. These experiments indicated that while using Hamming distances of 3 and 4 did not bring additional benefit in terms of identification robustness, using Hamming distance 2 allowed for the detection of false identifications. Taking into account the false identifications in the confidence function $f(s)$ from the previous section, allows the result of the error detection to be taken into account in the Bayesian update scheme. Thus, the Hamming code's ability to detect errors along with the Bayesian update scheme of the IDs probability resulted in less than 5% identification error on the UAV dataset described in Section 7. Furthermore, the Hamming code's ability could be used to learn the $f(s)$ from the actual data on-the-fly, improving the robustness of the identification even further.

6. EXPERIMENTS

To evaluate the performance of the proposed marker, we compared its localisation accuracy, detection range and identification reliability to state-of-the-art fiducial markers in a series of real experiments. Each of these tests used an RGB camera of an ASUS Xtion RGB-D sensor, as it corresponds with the type of sensor that is widely used on robotic platforms, providing a standard 640×480 image at 25 frames per second. This sensor was fixed to a FLIR E46-17.5 Pan Tilt Unit (PTU) which provided a ground truth for the marker position, orientation and velocity. This PTU was also mounted atop a mobile platform with a SICK s300 laser scanner. As the detectable range of the markers exceeds the range of a ASUS depth camera, the laser scanner with a range of up to 30m provided a reliable distance measurement that was also used for the ground truth in some of the experiments. To allow for a fair comparison of the proposed marker against the ARTAGS and APRILTAG, each of these markers were resized to occupy the same area of 3.817cm^2 .

A default calibration was also used, rather than specifically calibrating the camera, to demonstrate the system's performance in standard circumstances.

6.1 Detection and Identification Range



Figure 4: Illustration of range tests

The first test aimed to evaluate the effect that distance had on the performance of the system. The markers were affixed to the wall at a height equal to that of the camera. The mobile platform was then programmed to move backwards from a distance of 0.2 metres until the platform reached a distance of 7 metres from the wall. The movement occurred at a constant speed of 0.02 metres per second, which was selected in order to ensure that motion blur was not a significant factor.

Table 3: Maximum distances at which the markers were consistently detected and identified [m]

	WHYCON	APRILTAG	WHYCODE	ARTAGS
Detection	5.4	2.1	4.9	3.4
Identification	—	2.1	2.4	2.7

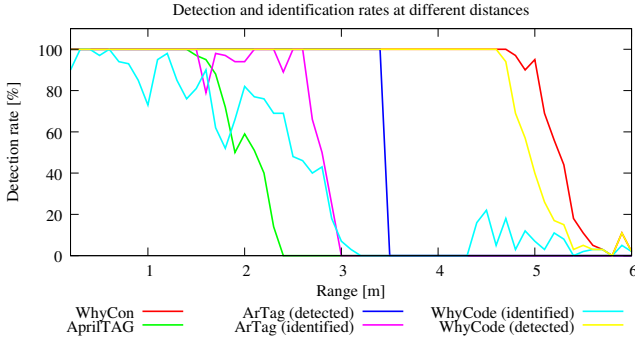


Figure 5: Maximum distances at which the markers were consistently detected and identified

As can be seen in Table 3 and Figure 5 the original WHYCON marker has proven to achieve the longest detection range of 5.4 metres. Although the WHYCODE marker was almost able to achieve a similar range, the new marker started to provide incorrect IDs once the distance had surpassed 2.4 metres. Similarly to that, the ARTAGS were undetectable at a range of 3.5 metres or more, and their correct identification was not reliable when the distance of the marker exceeded 2.7 metres. As for the APRILTAG, no incorrect IDs were reported. However, the distance at which the marker was reliably detectable was the lowest of the markers tested at only 2.1 metres.

6.2 Identification Range vs. Code Length

A similar test was also conducted on the WHYCODE marker to identify how changing the number of encoding bits affects the range at which the encoding can be correctly identified. As can be seen in Figure 6 using less than 8 bits for the code does not affect the range, while increasing it has a negative impact on the identification range. This corresponds with the expectation that the limiting factor of identification range is the size of the individual elements that make up the encoding pattern.

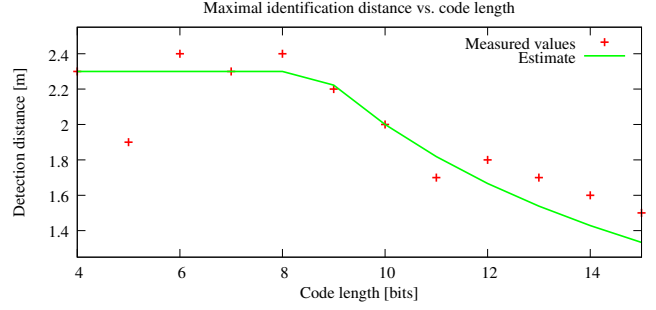


Figure 6: Dependence of maximal identification range on the Necklace code length n . The estimate is based on a formula $\min(2.4, 200/n)$

6.3 Robustness to Motion Blur



Figure 7: Illustration of motion blur tests

This test, which was intended to analyse the effect of motion blur on the markers, involved keeping the markers stationary whilst rotating the PTU. This setup not only ensured the equal movement of all the markers, but also created a stable, continuous and repeatable experiment which represented one of the system's intended applications: mobile robotic platforms with a moving on-board camera. With the markers affixed to the wall, the camera was placed exactly 1 metre from the wall and the PTU rotated from -90 degrees to +90 degrees at a constant speed. Figure 8 shows the speeds that were tested during this experiment with the resulting detection and identification ratios.

These results indicate that while both WHYCODE and WHYCON systems are less susceptible to motion blur, the APRILTAG identification scheme is more robust to motion blur compared to WHYCODE.

When attempting to decode the ID, the WHYCODE marker reported a number of incorrect results at the faster motions, which is caused by the fact that during these tests, the code did not employ any error detection or self-correction scheme. In contrast, the lexicographic error correcting [23] used by the APRILTAG meant that no incorrect IDs were ever detected during our tests.

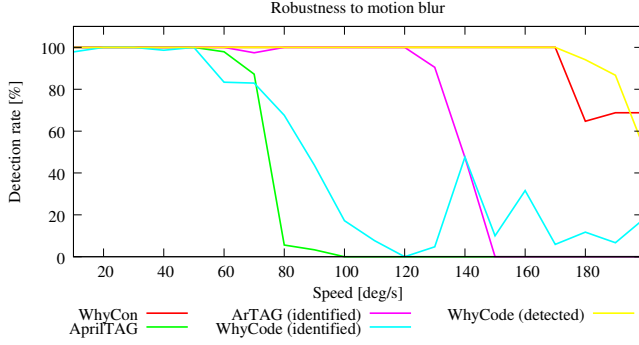


Figure 8: The results of the Motion Blur experiment - dependence of the detection rate on the marker velocity.

6.4 Accuracy of Angle Estimation



Figure 9: Rotation timelaps for angle estimation test

Since the x, y, z position estimation is identical to the original WHYCON method [13], which reports that its localisation accuracy is comparable to ARTAGS based markers, we tested only the accuracy of angle estimation. In contrast to the earlier experiments, the markers were this time placed on the robot's PTU which, whilst facing the free-standing stationary camera, used the pan and tilt functions to vary the angle of the markers. The recorded positions and rotations of the markers were then compared to the angle taken from the PTU. This comparison was then used to calculate an error rate for the system, see Table 4.

Table 4: Average error of angle estimates [radians]

	WHYCON	APRILTAG	WHYCODE	ARTAGS
Pitch/roll	0.024	0.023	0.020	0.038
Yaw	—	0.034	0.042	0.044

As can be seen from the above table, all markers exhibited average errors lower than 0.05 radians demonstrating that the system's ability to establish the marker's orientation was successful across all four systems. It should be noted that while the original WHYCON marker is unable to provide the yaw rotation, WHYCODE can estimate the yaw rotation with a high level of accuracy using the Necklace Encoding.

6.5 Robustness to Illumination Changes

The last test aimed to verify the performance of the system when subjected to various lighting conditions. To achieve this, the markers were positioned next to a large window in

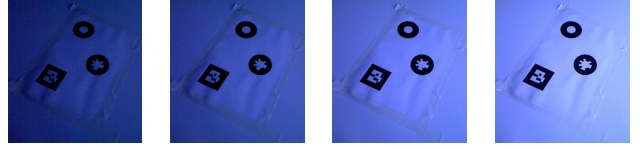


Figure 10: Illustration of lighting variations

order to utilise natural, ambient light and avoid the flickering sometimes caused by artificial light. By taking a photo every 10 seconds during the 25 minutes before and during sunrise, the markers were able to go from complete darkness to normal daytime lighting conditions. While the ARTAGS were detected in 64% of these images, APRILTAG, WHYCON and WHYCODE were detected in 71%, 72%, 74% of images respectively. Since the slight differences in performance may be attributable to slight variations in light, we can state that all the markers demonstrated a similar robustness to variable illumination.

6.6 Computational Complexity

In addition to the above tests, a number of computational performance tests were conducted on each of the systems. The first of these were conducted using procedurally generated images of size 5000×5000 pixels containing over 550 randomly placed markers. This test helped to evaluate each of the systems ability to handle, not only large images, but also images which contain high number of markers and varying levels of clutter. Although WHYCON and WHYCODE took more than a second to process the first frame, each subsequent frame was then processed significantly faster. The average time to process a single frame when comparing the APRILTAG and the WHYCODE systems can be seen in Table 5, which shows the main advantage of the WHYCODE method – its computational efficiency. Table 5 also shows that the identification and yaw estimation step do not slow down the original WHYCON method, which is two orders of magnitude faster than the ARTAGS and APRILTAG.

The performance boost WHYCON and WHYCODE results from the on-the-fly calculation of the detected segment statistics, which is naturally achieved by the flood-fill segmentation technique and which allows tracking without any computational overhead. Although the computational efficiency of both ARTAGS and APRILTAG could be improved by employing some tracking scheme, it is unlikely to achieve a two-orders of magnitude speed-up.

Table 5: Average processing time of an image with 550 markers [seconds]

Clutter	WHYCON	APRILTAG	WHYCODE	ARTAGS
none	0.06	3	0.06	3
little	0.07	16	0.07	14
large	0.07	15	0.07	15

6.7 Motion Prediction Speed Up

The computational performance of the system is affected by its ability to predict the approximate positions of the markers in the currently processed image. The earlier versions of the system simply searched for the pattern at the position it was detected in the last processed image, which was sufficient for experiments, where the markers moved only slowly. However, if the system is deployed in situations, where the tracked objects move rapidly, e.g. when the system is used to track aerial robots [19], the aforementioned method failed to predict the positions properly, which resulted in system slowdown. Thus, we improved the motion prediction by taking into account the marker velocity as described in Section 4.1.

Table 6: Average processing time of sequences with rapidly-moving markers [ms]

Tracking type	Sequence number			
	I	II	III	IV
Position only	21.5	24.1	25.2	30.1
Position + velocity	19.7	20.2	23.0	28.3

To evaluate the effect of the improved tracking scheme, we calculated the average time it takes to process an image of the UAV dataset described in Section 7, which contains four sequences of flying robots captured from an UAV-mounted camera. The Table 6 shows that taking into account the tracked pattern velocity reduces the computational load by ~10-20%.

7. DATASETS

Another contribution of this work is the addition of two publicly available datasets that ensure the repeatability and verification of the results obtained above. The first of which provides numerous videos, along with position and movement statistics taken from a SCITOS robot, which can be used to test the abilities of the WHYCODE, WHYCON, APRILTAG and ARTAGS systems that are mentioned above. In addition to the benchmarking dataset, a real world example of the new system being utilised to track drones flying outside, along with their reported positions taken from the drones IMU and GPS devices. Both of these datasets are available for download from: <http://lncn.eu/fidusets>.

8. CONCLUSION

In this paper, we present an extension to the marker used by the WHYCON tracking system. The proposed method not only utilises a new encoding algorithm which allows identification of each marker, but also extends the system to allow the full localisation of a marker with 6 DOF. Furthermore, we introduce several improvements that strengthen the robustness, accuracy and computational efficiency of the detection and identification. By keeping the simple roundel design, the proposed marker is not only backwards compatible with the previous system, but also maintains its sub-pixel

(2D) and millimetre (3D) precision, and high computational efficiency.

The results of our study show that the WHYCODE system, despite the additional overhead of having to decode marker IDs, performed similarly to the original WHYCON system and outperformed the comparative systems in both accuracy and speed. By exceeding the high level of performance demonstrated by the APRILTAG and ARTAGS, and at two orders of magnitude faster, the proposed system achieves a strong level of accuracy without the high computational requirements. These achievements therefore make the proposed system particularly applicable to resource-constrained systems and scenarios, where the reliable and swift tracking of multiple robots is a necessity. Moreover, the WHYCON system can reliably detect smaller markers at longer ranges, which is also makes it a popular alternative to APRILTAG or ARTAGS.

In the future, we will explicitly model uncertainty of the marker locations, which should not only improve our system’s accuracy [17], but also its coverage by allowing to fuse input from multiple cameras.

Acknowledgements

The work is partially supported by the European Community’s funded projects 600623 ‘STRANDS’ and 732773 ‘ILIAD’, the Research Investment Fund (RIF) of the University of Lincoln, and the Czech Science Foundation project No. 17-27006Y.

9. REFERENCES

- [1] D. Afolabi, K. L. Man, H.-N. Liang, S.-U. Guan, and T. Krilavičius. 1543. monocular line tracking for the reduction of vibration induced during image acquisition. *Journal of Vibroengineering*, 17(2), 2015.
- [2] F. Arvin, T. Krajník, A. E. Turgut, and S. Yue. COS- ϕ : artificial pheromone system for robotic swarms research. In *IROS*. IEEE, 2015.
- [3] F. Arvin, A. E. Turgut, T. Krajník, and S. Yue. Investigation of cue-based aggregation in static and dynamic environments with a mobile robot swarm. *Adaptive Behavior*, 2016.
- [4] L. Batten et al. Classification of Chemical Compound Pharmacophore Structures. 1999.
- [5] W. Y. Chen and J. D. Louck. Necklaces, mss sequences, and dna sequences. *Advances in applied mathematics*, 18(1):18–32, 1997.
- [6] D. L. De Ipiña, P. R. S. Mendonça, and A. Hopper. TRIP: A low-cost vision-based location system for ubiquitous computing. *Personal and Ubiquitous Computing*, 2002.
- [7] C. Feng. *Camera Marker Networks for Pose Estimation and Scene Understanding in Construction Automation and Robotics*. PhD thesis, University of Michigan, 2015.
- [8] M. Fiala. Artag, an improved marker system based on artoolkit. *National Research Council Canada, Publication Number: NRC*, 2004.

- [9] R. Forster. Manchester encoding: opposing definitions resolved. *Engineering Science & Education Journal*, 9(6):278–280, 2000.
- [10] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.
- [11] C. Hu, F. Arvin, C. Xiong, and S. Yue. A bio-inspired embedded vision system for autonomous micro-robots: the lgmd case. *IEEE Transactions on Cognitive and Developmental Systems*, PP(99):1–1, 2016.
- [12] T. Krajník, M. Nitsche, J. Faigl, T. Duckett, M. Mejail, and L. Přeucil. External localization system for mobile robotics. *Proceedings of the 16th International Conference on Advanced Robotics (ICAR 2013)*, pages 1–6, 2013.
- [13] T. Krajník, M. Nitsche, J. Faigl, P. Vaněk, M. Saska, L. Přeucil, T. Duckett, and M. Mejail. A practical multirobot localization system. *Journal of Intelligent & Robotic Systems*, 76(3-4):539–562, 2014.
- [14] M. Kulich, J. Chudoba, K. Košnar, T. Krajník, J. Faigl, and L. Přeucil. Syrotek distance teaching of mobile robotics. *IEEE Transactions on Education*, 2013.
- [15] D. Lehmer. On Euler’s totient function. *Bulletin of the American Mathematical Society*, 38(10):745–751, 1932.
- [16] P. Lightbody, T. Krajník, and M. Hanheide. A versatile high-performance visual fiducial marker detection system with scalable identity encoding. In *Proceedings of the Symposium on Applied Computing, SAC ’17*, pages 276–282, New York, NY, USA, 2017. ACM.
- [17] R.-G. Mihalyi, K. Pathak, N. Vaskevicius, T. Fromm, and A. Birk. Robust 3d object modeling with a low-cost rgbd-sensor and ar-markers for applications with untrained end-users. *Robotics and Autonomous Systems*, 66:1–17, 2015.
- [18] E. Olson. AprilTag: A robust and flexible visual fiducial system. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3400–3407. IEEE, May 2011.
- [19] M. Saska, T. Baca, J. Thomas, J. Chudoba, L. Preucil, T. Krajník, J. Faigl, G. Loianno, and V. Kumar. System for deployment of groups of unmanned micro aerial vehicles in gps-denied environments using onboard visual relative localization. *Autonomous Robots*, 41(4):919–944, 2017.
- [20] J. Sawada. Generating Bracelets in Constant Amortized Time. *SIAM Journal on Computing*, 31(1):259, 2001.
- [21] M. C. Silverman, D. Nies, B. Jung, and G. S. Sukhatme. Staying alive: a docking station for autonomous robot recharging. In *Robotics and Automation, 2002. Proceedings. ICRA ’02. IEEE International Conference on*, volume 1, pages 1050–1055 vol.1, 2002.
- [22] P. technologies. Phoenix 3d motion capture.
- [23] A. Trachtenbert. Computational methods in coding theory. Master’s thesis, University of Illinois at Urbana-Champaign, 1996.
- [24] Vicon. Vicon MX Systems.
- [25] D. Wagner and D. Schmalstieg. Artoolkitplus for pose tracking on mobile devices. In *Proceedings of the 12th Computer Vision Winter Workshop (CVWW’07)*, February 2007.
- [26] S. Yang, S. A. Scherer, and A. Zell. An onboard monocular vision system for autonomous takeoff, hovering and landing of a micro aerial vehicle. *Journal of Intelligent & Robotic Systems*, 69(1-4):499–515, 2013.