Designing a Marker Set for Vertical Tangible User Interfaces

Geoffrey Attard
Faculty of Science and Technology
Middlesex University Malta
Pembroke, Malta
geoffreyanthonyattard@gmail.com

Clifford De Raffaele, *IEEE Member*
Faculty of Science and Technology
Middlesex University Malta
Pembroke, Malta
cderaffaele@ieee.org

Serengul Smith
Faculty of Science and Technology
Middlesex University London
Hendon, UK
s.smith@mdx.ac.uk

***Abstract*—Tangible User Interfaces (TUI)s extend the domain of reality-based human-computer interaction by providing users the ability to manipulate digital data using physical objects which embody representational significance. Whilst various advancements have been registered over the past years through the development and availability of TUI toolkits, these have mostly converged towards the deployment of tabletop TUI architectures. In this context, markers used in current toolkits can only be placed underneath the tangible objects to provide recognition. Albeit being effective in various literature studies, the limitations and challenges of deploying tabletop architectures have significantly hindered the proliferation of TUI technology due to the limited audience reach such systems can provide. Furthermore, available marker sets restrict the placement and use of tangible objects since if placed on top of the tangible object, the marker will interfere with the shape and texture of the object limiting the effect the TUI has on the end-user. To this end, this paper proposes the design and development of an innovative tangible marker set specifically designed towards the development of vertical TUIs. The proposed marker set design was optimized through a genetic algorithms to ensure robustness in scale invariance, the capability of being successfully detected with distances of up to 3.5 meters and a true occlusion resistance of up to 25%, where the marker is recognized and not tracked. Open-source versions of the marker set are provided through research license on www.details_left_intentionally_empty_for_review.edu**

Keywords—Vertical Tangible User Interface, Human-Computer Interaction, Computer Vision, Computer Generated Markers

I. INTRODUCTION

Over recent years, TUI technology has gained popularity in many different fields. The most popular TUI architecture is the tabletop form, which uses a semi-translucent surface to allow a projector to project from underneath the table and diffuse the graphics on the table surface [1], [2]. The semi translucency is also important to allow a camera inside the table to have a clear view of any markers placed flat on the table surface. The software part of the system includes two different parts of applications, which implement a one-way communication between them. The first system is a toolkit used to recognize the orientation and position of a set of markers placed on top of the table. The second system is specifically designed and developed to show various animations on the table surface via the projector in response to marker movement. In most cases, the second system is developed for a single TUI application. On the contrary, the toolkit is used for all the different graphical system implementations.

A TUI can include various different audio, visual and tactile feedback elements, this results in various sensory engagements which overall assist the user to better understand and envision the concept being delivered by the TUI system [3]. Furthermore, a TUI system promotes collaborative working and learning [4]. In the education sector, it was observed that a TUI also attributes to better knowledge retention when compared to wimp interfaces [1]. This technology gained popularity because of the smooth transition between physical and digital interaction enabling the end-user to manipulate digital data by moving tangible objects. This aspect of a TUI system is based on the MCRpd human-computer interaction model [5].

Not every aspect of a TUI is positive though, the implementation part of a TUI system tends to be challenging. Current marker recognition toolkits come with various difficulties during and after the implementation of a TUI system [6]. Such difficulties are holding the acceptance of such an advantageous technology. Implementation complications include the high procurement expenses and the significant footprint needed by a tabletop TUI which in some cases constrains the number of places the TUI can be installed in [2]. Furthermore, in most cases, technical expertise is needed before and during the operation of the TUI system limiting the wide adoption of such systems. The adoption of a vertical TUI architecture [7], aims to mitigate some difficulties experienced with TUI tabletops. Unfortunately, however, this experimental architecture made use of a bulky back-projection configuration due to the type of markers employed within the setup. The utilized marker set also hindered the exposure and use of tangible objects within the setup, undermining its effectiveness to exploit the inherent effectiveness of TUI elements.

To this end, in order to eliminate all the aforementioned TUI problems, a new marker set was developed to enable the development of vertical TUIs which virtually eliminate the procurement expenses for dedicated hardware, but rather uses a conventionally available whiteboard – projector – camera setup as defined in [8]. Although many marker technologies already exist, none of them can be used in a true vertical TUI setup. Existing markers will either interfere with the shape and texture of the tangible object while others cannot be effectively recognized from the required whiteboard camera distance.

II. LITERATURE REVIEW

In image processing a marker recognition algorithm can be characterized by the following strengths; marker recognition rate, perspective resistance, inter-marker distance, marker occlusion resistance, marker scale invariance, illumination invariance, minimal false positive detection and marker jittering resistance. A method to recognize markers in suboptimal lighting conditions was introduced in [9] where it was demonstrated that marker edges can be estimated. To recognize a partly occluded marker however, the rebuild of the

whole marker can only be done from the partial visible data [10]. Experiments with Artoolkit showed that some additional factors may negatively affect the recognition rate of this engine, such as the quality of the camera used for tracking, the marker printing quality, the smoothness of the surface used and the complexity of the objects surrounding the marker [11].

As illustrated in Fig. 1, the most popular frameworks in current literature employ bitonal (black and white) markers for better contrast ratios which leads to improved detection rates [12][13][14][15][16]. Furthermore, the 2D planar markers illustrated in Fig. 1(a-d), apart from identification they were also designed to calculate their pose estimation, which has seen wide adoption in augmented reality, photogrammetry and computer vision applications.

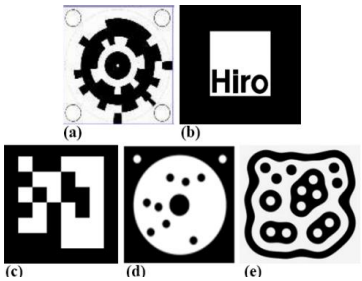

Fig. 1: Sample marker sets from current literature toolkits including: (a) Artoolkit, (b) Trackmate, (c) ARTag, (d) X-Tag, (e) reacTIVision

In difference to the planar markers illustrated above, Fig. 2 depicts the currently available circular markers in literature. The marker in Fig. 2(a) was proposed initially proposed in [17] and further improved in [18]. This type of marker pattern and the proposed detection system showed robust results in recognition rate and occlusion resistance. Unfortunately, however, this marker was not designed for rotation detection and the circular pattern limits the number of markers that can be generated without affecting the inter-marker distance. Conversely, the fiducial marker depicted in Fig. 2(b), was designed for multiple usages, one of which is pose estimation [19]. Whilst concluding that this marker has high occlusion resistance and is robust against different types of noise [20], the intrinsic nature of the marker design pose a restrictive limit on the distance from which it can be recognized and tracked due to the small size of the dots used and inherent data complexity.

As a marker can be occluded or distorted by many factors during physical operation, a solution to such a problem was proposed in [21], whereby estimation on the marker location during brief periods of marker occlusions where mitigated by a Kalman filter [22]. Given the nature of the Kalman filter adopted, the marker pose data was also smoothened. As concluded from experimental tests, the tracking approach undertaken by TUI toolkits such as Reactivision [13] on markers shown in Fig. 1(e), poses a further limitation since it was observed that each fiducial can be partly occluded during the tracking state only and is not recognized at all when it is presented in front of the camera already occluded.

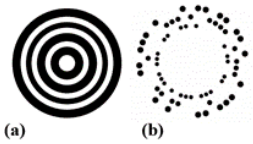

Fig. 2: (a) Concentric Circle Marker (b) RUNE-Tag Marker

The aforementioned marker sets have all reported good recognition rates when used in their intended environment involving little to no occlusions. Unfortunately, however, none of the current markers shows applicability towards the unique requirements for effective usage in a vertical TUI setup, since such markers need to have enough space for a tangible object to be placed on top of them without occluding their unique data pattern. Apart from this, the marker needs to have enough clusters of "pixels" that it can be easily recognized from a distance of 2.5 to 3 meters given the contextual use and setup of vertical surfaces. Furthermore, in a vertical TUI setup, each marker has to be recognized even if a part of it is already occluded when presented in the capture frame. This is because in a vertical TUI setup as shown in Fig. 3, marker occlusions are greatly increased, occlusions such as the tangible object or its shadow.

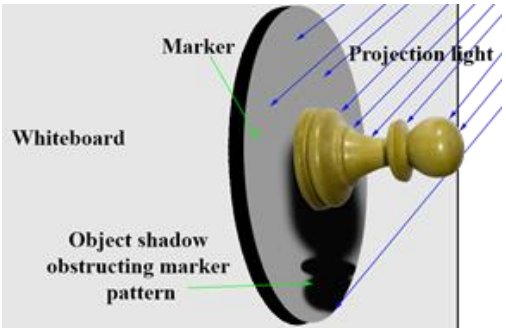

Fig. 3: Marker Occluded by the Tangible Object's Shadow

When further taking users into account, marker occlusions are increased drastically because the user may occlude a significant part of the marker just by standing in front of it or by simply moving the object. This implies that required marker set cannot reliably rely on tracking in such a scenario but should alternatively incorporate sufficient information in its design for true occlusion resistance. In contrast, current circular markers which may be used in this type of configuration are either too complex to be detected from the required distance while using a reasonable marker size or lack some important requirement such as rotation detection. This led to the need of a new marker design which retains its patter visible whilst providing enough in order to enable the placement of a tangible object on top of it without interfering with the visual shape and texture of the tangible object itself.

Moreover, based on the requirements elicited from recognition engines currently available in literature [8], [13], [16], markers with a sudden change in contrast and multiple edges will help improve the recognition rate. In the case of a vertical TUI, such factors will improve marker recognition rate even though a part of it is occluded by the object's shadow and/or the user's hand. To counteract such big occlusions each marker pattern has to be unique all around so when the algorithm detects the visible part of that particular marker it won't recognize it as a false positive. Whilst the amount of data in each marker has to be enough for marker uniqueness and pose information, this pattern complexity is concurrently limited by the recognition distance between the marker and the capturing setup. This unique criterion thus presents an optimization tradeoff for marker sets to be suitable in Vertical TUI setup adoptions which are further compounded by a scalability requirement for different sized markers which enable marker size adaption dictated by different tangible object sizes.

III. MARKER DESIGN

To limit the complexity and burden in adopting the proposed marker set, a simple passive marker design was developed that could be commonly printed on paper and required minimal hardware to recognize/operate. In line with literature, in order to reach good recognition rates, the markings needed to embed various sudden changes in contrast and present multiple sharp edges in different sizes. A salient change between the pixels and the background will also help increase the recognition rates, so a bitonal (black and white) color was chosen. Given that markers need to be visible from the front, the most appropriate shape identified was that of a circular structure with a white space in the middle for placement of the tangible object. Other shapes will simply use more scene space or limit more the space for the recognition pattern.

The pattern to be recognized had to be big enough that it can be detected using a mid-range webcam or a mobile device. At the same time, the overall marker needed to be small enough to keep the marker at a reasonable size and not occupy significantly the vertical interface. To this end, a scale invariant design was adopted whereby the scale of each pixel and the white space between them would vary depending on the entire size of the marker. This allowed for marker scaling, which could be changed according to the physical size of the tangible object placed on top of it. Moreover, this scale invariance was designed to retain a recognition rate for conventional projection/whiteboard setups at a distance of 2.5 to 3 meters away.

Preliminary pattern testing was performed using digital images. The pattern tested was searched for features sets and a score was accredited depending on the number of features and the inter-marker distance between the sets. The best pattern recorded was that of containing and manipulating data within three concentric circles placed within each other. Subsequently, an experimental pattern testing was carried out by printing and placing the pattern to be tested on a white foreground, so only the pattern features are detected. All markers were printed using a laser printer set to 600 dots per inch (DPI). Each marker was then tested twice using a mid-ranged web camera and a mobile phone used as a camera which were placed at various distances. Apart from the number of features found and the distance between each feature, the distance between the marker and the camera was also used to determine the recognition quality of each experimental pattern.

Following the empirical knowledge obtained from these experiments, the final marker version was designed with three individual circles made up of 21 square pixels each resulting in a 63 pixel marker. As depicted in Fig. 4, each pixel was widened vertically to overlap on the other circles in order to create various distance variations between each sharp edge. This design produced not less than 100 features per marker. The resulting markers were well recognized from a 3.5 meter distance and features were distributed all around the marker with good distance between them. The amount of pixels allows 4,096 different marker combinations that were later reduced to 300 in order to make each marker unique all around the circle pattern for better occlusion resistance and a bigger inter-marker space.

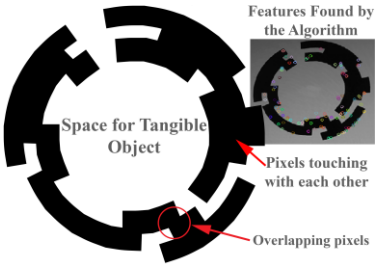

Fig. 4: Final Marker Design

IV. GENERATING THE MARKER SET

A. Marker Generation Algorithm

To systematically create an optimal set of unique markers, a genetic algorithm was used whereby marker designs were represented inside the algorithm by an array of binary values. Each value represents one pixel, with 1 implying a colored or black pixel whilst 0 representing blank space or white pixels. The fitness function was designed to compare each individual marker with the rest of the population and calculate the frequency of similar groups of N pixels. The two markers with the least amount of similarities were then selected for iterative reproduction and the marker, which has the most same groups of pixels, is replaced by the offspring. To maintain diversity within the whole population, each offspring is mutated based on a random variable.

This strategy ensured that the algorithm prevented premature convergence within a local minima. Another validation was implemented to control the early random creation of the population were each group of N pixels are compared with the whole population in a sliding window manner before they are accepted and added to the population multi-dimensional array. Comparison of pixels was conducted to imitate the real life comparison of the marker pattern and thus the binary array representing the marker pixels was be divided into distinct rows and columns for an effective comparison with the binary data of other markers as illustrated in Fig. 5. This methodology took into account the manner in which popular computer vision toolkits operate and thus aimed to reduce the potential false positive rate when different features of the complex patterns are detected in each frame.

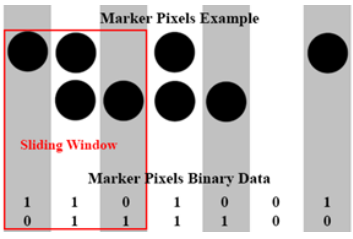

Fig. 5: Binary Data Converted to Marker Pattern

B. Fitness Function

As illustrated in Fig. 6, the fitness function was developed to use a sliding window of 3 pixels. Two sliding windows were correspondingly adopted on each paired comparison. So as to account for each possible placement of the circular marker, the sliding windows were iteratively rotated through full revolutions, assessing the pixel similarity at each rotational increment by 1 pixel (17.143 degrees). This process was subsequently conducted to directly compare each

considered marker with the remaining marker set population consecutively.

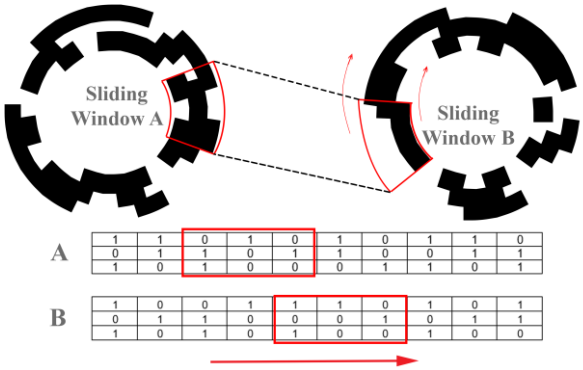

Fig. 6: Sliding Window Fitness Function

To this end, the fitness function was able to quantify the amount of sliding window matches a single marker has with the entire population. Markers with the least amount of matches are chosen to produce a new offspring which replaces the chromosome with the biggest amount of matches.

C. Maintaining Diversity within the Population

The crossover method used to generate an offspring takes genes from both of the parents which are randomly selected to create the chromosome for the offspring. To maintain diversity within the population and prevent premature convergence of the algorithm, after the crossover function, each offspring is mutated by randomly selecting and inverting up to 30 different genes. Fig. 7 illustrates the crossover function logic in more detail.

```
Loop through the first chromosome's genes
for (int i = 0; i < parentOne.length; i ++) {
    Generate a random binary number
    rndNum = getRandomNumberInts(0, 1);
    if (rndNum == 0) {
        if random number is equal to 0 copy the first chromosome
        current gene to the offspring's current gene
        child[i] = parentOne[i];
        if(i+1 < parentOne.length){
            If end of first chromosome is not reached copy the
            next gene to the offspring's next gene
            i ++;
            child[i] = parentOne[i];
        }
    } else {
        if random number is equal to 1 copy the second chromosome
        current gene to the offspring's current gene
        child[i] = parentTwo[i];
        if(i+1 < parentTwo.length){
            If end of second chromosome is not reached copy
            the next gene to the offspring's next gene
            i ++;
            child[i] = parentTwo[i];
        }
    }
}
```

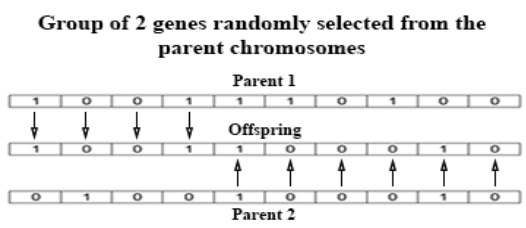

Fig. 7: Crossover Approach

D. Convergence of the Algorithm

The genetic algorithm reaches a global minimum when the amount of most similar pixels remained consistent for 500 generations. This threshold was found by reducing the population number by 50 each time the genetic algorithm converges to a global minimum. The first population amount was 700 where it reached the global minima with a maximum number of 400 matches, so the population was iteratively reduced by 50 and followed by another genetic algorithm execution until an acceptable number of matches was reached.

E. Rendering the Marker Set

After the genetic algorithm converges, each binary array representation is automatically converted to illustrate a pixel value within the marker image. Each pixel was created using the Java Arc2D class, whereby the width and height of each pixel were set by adjusting the arc stroke. Finally, the marker was smoothened by parsing through an antialiasing image filter. After calculating and setting the location of each group (circle) of pixels the arcs had to be evenly spaced from each other, this was achieved by dividing equally dividing the circular pattern by 21 pixel positions, and altering the arc angle position as illustrated in Fig. 8.

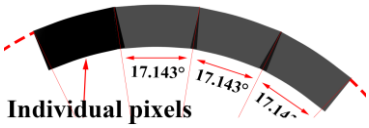

Fig. 8: Marker Pixel Placement

For each marker, a high-resolution buffered image was created and set with a size of 2100x2100. The substantial size allowed the scaling to various different marker sizes to later be printed in high fidelity. Java 2d graphics were then created from the buffered image and used to draw the pixels according to the binary data arrays generated by the genetic algorithm.

V. Marker Set Filtering

A. Correlation between Markers

To minimize the potential rate of false positives on the generated set of 300 markers, these candidate elements were further filtered according to their respective Hamming distance. Since Hamming distance calculations are not rotationally invariant, each marker was progressively rotated by 17.143 degrees (the pixel width) and each time compared with the remaining population until it reached a full 360-degree rotation. The smallest/worst hamming distance results with each rotation were recorded and finally, the hamming distance value of each individual marker was used to rank the markers from least similar to most similar. Fig. 9 depicts the comparison process adopted for each marker which was consecutively undertaken the rest of the population. The obtained results from this analysis are illustrated in Fig. 10 where unique markers are ranked according to their hamming distance.

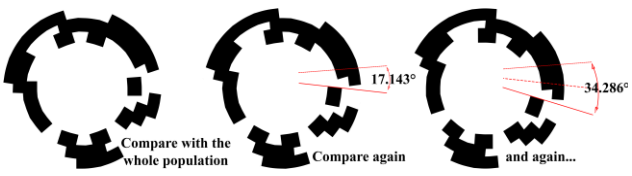

Fig. 9: Rotated Hamming distance Markers Comparison

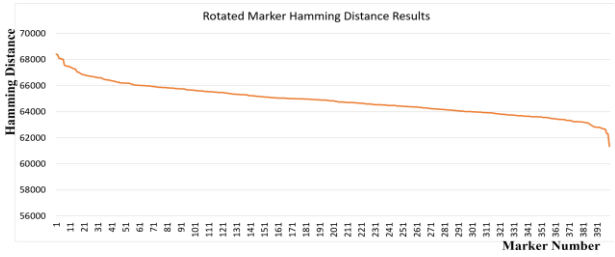

Fig. 10: Hamming Distance Results for the population marker set

B. Recognition Engine Marker Filtering

To further refine and optimize the selection of the complete marker set, a second filter was developed to assess each marker for its occlusion resistance whilst retaining no false positive detection. This test was designed to faithfully emulate an extreme user interaction process with the TUI marker where, as illustrated in Fig. 11(a), approximately 25% of the marker is occluded by the user's hand. This implies that from a recognition perspective, such occlusion might provide a set of different features each time a frame is processed. The technique chosen to elicit the best markers was to load the consecutively load markers in accordance with the previously ranked hamming distance and display them in a window while using an obstruction to occlude 25%. To account for the circular nature of the marker, the occlusion was rotated in the form of an animation which gradually covers the marker through a 360° rotation as illustrated in Fig. 11(b). A template matching approach was then used to recognize each marker while keeping a record on each marker recognition rate and false positives as shown in Fig. 11(c).

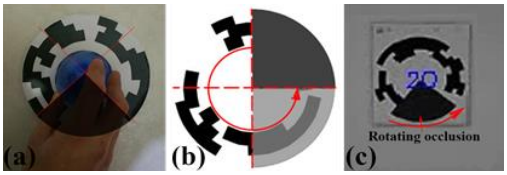

Fig. 11: (a) Hand Occlusion (b) Rotating occlusion path (c) Rotating occlusion animation

The occlusive animation was executed for a duration of 19 seconds till it revolved around the entire marker. Based on a capturing and processing rate of 30 frames per second (FPS), this methodology processed 570 frames during the occlusion animation on each marker, ensuring the sufficient capture of different potential feature vector sets. A frequency count was retained on the number of successfully recognized frames within this sample as well as a record of false positives generated on each marker. The results for the recognition rate amongst the entire marker population is illustrated in Fig. 12.

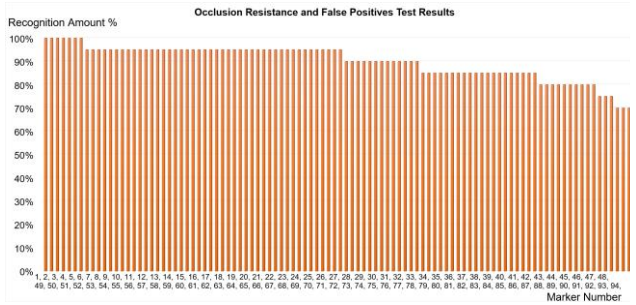

Fig. 12: Occlusion resistance and false positives test results

So as to establish a threshold criterion on the amount of occlusion resistance available through the designed marker set, a progressive test sequence was adopted, whereby in each iteration the amount of occlusion applied across each marker was increased. This test was performed by increasing the animation occlusion by 3.6 degrees (1%) after each marker set test, and iterated until the recognition rate was decreased below 75%, at which point the functionality was deemed practically unusable. The results illustrated in Fig. 13 were collected by averaging each test results. It was concluded that the system was effectively usable up to a maximum occlusion of 31%, where an 89.54% recognition rate was achieved.

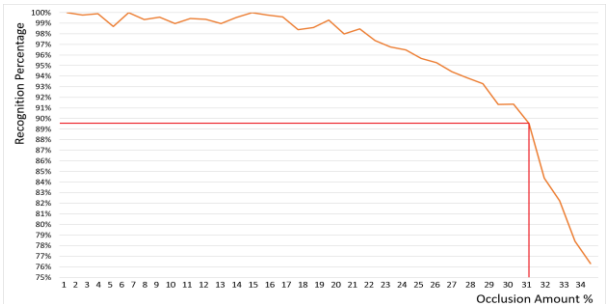

Fig. 13: Maximum Marker Occlusion Test

VI. Testing and Evaluation

In line with common approaches adopted by recognition engines, the proposed marker set was evaluated through the adoption of a SIFT feature detection algorithm [23] and classified using a k-nearest neighbor classifier. This methodology was developed to provide an objective analysis of the strengths and limitations of the proposed designs, enabling the gathering of results on uniqueness, recognition from different distances and marker scale invariance. During marker evaluation, the web camera parameters were left unchanged capturing at 2.1 megapixels (MP)s running at 30 FPS. A second test was later executed using a mid-range smartphone device (Samsung Galaxy Note 4) with a camera capturing capability at 16 MP at 30 FPS. For data consistency, each marker was printed with a laser printer set to 600 Dots per Inch (DPI) and each captured image was digitally cropped onto a whiteboard area of 1.8m x 1.2m.

A total of 14 markers were tested starting from an 8cm diameter to a maximum of 17cm with an incremental difference in diameter of 1cm between each marker. The camera distance from the marker in test ranged between 1.6m to 3.5m, spanning between the minimum distance needed to capture the entire whiteboard to the maximum distance depending on the whiteboard size and the projector distance. The test at each position was set to last 10 seconds; during this time a total of 300 frames (30FPS x 10 seconds) were captured for each marker. The recognition rate with respect to capture distance and marker size is illustrated in Fig.14.

The space available for the tangible object on top of each marker in cm² is half of the marker diameter size, e.g. the space available on a 13cm marker is 6.5cm². To this end, practical constraints of common physical objects imply that most common adoptions will utilize 13-16cm markers providing an area of 6.5cm²-8cm² of tangible placement area respectively. A potential limitation of this marker set is that some objects which due to their appearance and form factor, might additionally challenge a marker due to excessive occlusion or unplanned object features.

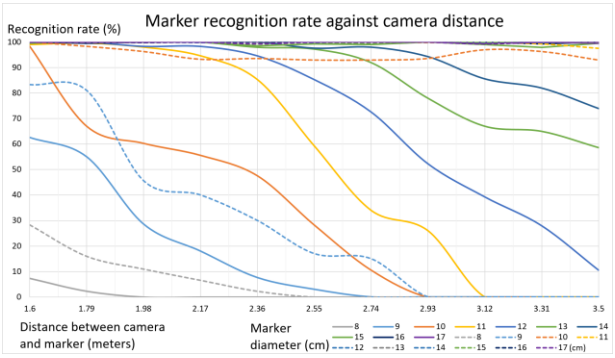

Fig. 14: Recognition rate of different marker scales against camera distance. Straight lines denote capture from a 2.1MP camera and dotted lines represent results from a 16 MP smartphone camera.

Results illustrated in Fig. 14 concluded that a mid-ranged web camera with a 2.1MP sensor had a 98% recognition rate on markers with a diameter between 13cm and 14cm from a maximum distance of 2.5m. Markers with a 15cm diameter and greater had a 100% recognition rate from distances ranging between 1.6m to 3.5m. More effectively, the 16MP smartphone sensor had a 93% recognition rate on markers with a 10cm diameter from 2.5m distance. Moreover, markers with an 11cm diameter and greater were recognized at over 97.6% of the time from distances ranging between 1.6m to 3.5m.

The marker set was also tested for false positives by exposing each marker for two minutes from a typical whiteboard projector setup distance of 2.9m. For data consistency, during this test, the image was digitally cropped on the whiteboard area. This test concluded that the proposed marker set has an average true positive rate of 98.99% and an average false positive rate of 1.6325%. This test was executed with true recognition and no prediction algorithms were used.

VII. Conclusion

This paper presents a new marker set that aims to directly alleviate TUI implementation barriers currently faced by developers and users of TUI tabletop systems. The proposed marker set reintroduces vertical TUI setups as an attractive alternative due to their inherent simplistic hardware and calibration setups. Such systems also eliminate the need and expenses incurred into building custom-made tabletop architectures which require constant maintenance and hardware/software calibrations. The attractive results gained by the proposed marker set outline a true positive rate of 98.99% and a false positive rate of 1.6325% making it possible to develop vertical TUIs with robust real-time recognition. The proposed marker set is intrinsically designed for scale invariance, allowing the provision and detection of pose information from capturing distances of up to 3.5 meters [8]. Furthermore, through the use of optimizing algorithms and filters, markers are able to guarantee a true occlusion resistance of at least 25% during operation, making the setup robust to vertical TUI interactions.

References

[1] C. De Raffaele, S. Smith, and O. Gemikonakli, "The Application of Tangible User Interfaces for Teaching and Learning in Higher Education," Innovative Teaching and Learning in Higher Education, no. January, pp. 215–226, 2017.

[2] O. Shaer, N. Leland, E. H. Calvillo-Gamez, and R. J. K. Jacob, "The TAC paradigm: Specifying tangible user interfaces," Personal and Ubiquitous Computing, 2004.

[3] J. Quarles, S. Lampotang, I. Fischler, P. Fishwick, and B. Lok, "Tangible User Interfaces compensate for low spatial cognition," in 3DUI - IEEE Symposium on 3D User Interfaces 2008, 2008.

[4] E. B. Susman, "Cooperative Learning: A Review of Factors That Increase the Effectiveness of Cooperative Computer-Based Instruction," Journal of Educational Computing Research, 2005.

[5] H. Ishii, "Tangible Bits: Beyond Pixels," Proceedings of the Second International Conference on Tangible and Embedded Interaction (TEI'08), Feb 18-20 2008, Bonn, Germany, 2009.

[6] T. Wallbaum, A. Matviienko, W. Heuten, and S. Boll, "Challenges for designing tangible systems," CEUR Workshop Proceedings, vol. 1861, pp. 21–23, 2017.

[7] A. Ricardo and J. Corredor, "VeRITable:Tangible User Interface - YouTube," 2009. [Online]. Available: https://www.youtube.com/watch?v=_8O38qV42f8. [Accessed: 02-Mar-2017].

[8] G. Attard, C. de Raffaele, and S. Smith, "TangiBoard: A Toolkit to Reduce the Implementation Burden of Tangible User Interfaces in Education," in 2019 IEEE Advanced Information and Communication Technologies, 2020, pp. 1–7.

[9] V. Agnus, S. Nicolau, and L. Soler, "Illumination independent marker tracking using cross-ratio invariance," Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology - VRST '14, vol. 35, no. 5, pp. 22–33, 2014.

[10] M. Fiala, "Designing Highly Reliable Fiducial Markers," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 7, pp. 1317–1324, 2010.

[11] D. Khan, S. Ullah, and I. Rabbi, "Factors affecting the design and tracking of ARToolKit markers," Computer Standards & Interfaces, vol. 41, pp. 56–66, 2015.

[12] J. Y. Tinevez et al., "TrackMate: An open and extensible platform for single-particle tracking," Methods, vol. 115, pp. 80–90, 2017.

[13] R. Bencina and M. Kaltenbrunner, "The Design and Evolution of Fiducials for the reacTIVision System," Proc. 3rd International Conference on Generative Systems in the Electronic Arts, no. January, 2005.

[14] T. Birdal, I. Dobryden, and S. Ilic, "X-Tag: A fiducial tag for flexible and accurate bundle adjustment," Proceedings - 2016 4th International Conference on 3D Vision, 3DV 2016, pp. 556–564, 2016.

[15] M. Fiala, ARTag, a Fiducial Marker System Using Digital Techniques, vol. II. IEEE, 2005, pp. 590–596.

[16] ARToolworks, "Artoolkit," Marker Tracking and HMD Calibration for a video-based Augmented Reality Conferencing System, 1999. [Online]. Available: http://www.hitl.washington.edu/artoolkit/. [Accessed: 12-Mar-2019].

[17] L. Calvet, P. Gurdjos, and V. Charvillat, "Camera tracking using concentric circle markers: Paradigms and algorithms," Proceedings - International Conference on Image Processing, ICIP, pp. 1361–1364, 2012.

[18] L. Calvet, P. Gurdjos, C. Griwodz, and S. Gasparini, "Detection and Accurate Localization of Circular Fiducials under Highly Challenging Conditions," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 562–570, 2016.

[19] F. Bergamasco, A. Albarelli, E. Rodola, and A. Torsello, "RUNE-Tag: A high accuracy fiducial marker with strong occlusion resilience," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 113–120, 2011.

[20] F. Bergamasco, A. Albarelli, L. Cosmo, E. Rodola, and A. Torsello, "An Accurate and Robust Artificial Marker Based on Cyclic Codes," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 12, pp. 2359–2373, 2016.

[21] H. C. Kam, Y. K. Yu, and K. H. Wong, "An improvement on ArUco marker for pose tracking using kalman filter," Proceedings - 2018 IEEE/ACIS 19th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, SNPD 2018, pp. 65–69, 2018.

[22] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," Journal of Basic Engineering, vol. 82, no. 1, p. 35, 2011.

[23] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, vol. 60, no. 2, pp. 91–110, 2004.