https://doi.org/10.2352/ISSN.2169-4451.2021.37.73
©2021 Society for Imaging Science and Technology

Atomic growing for grid alignment*

Yujian Xu[⋆] **Matthew Gaubatz**[†] **Stephen Pollard**[†] **Robert Ulichney**[⋆] **Jan Allebach**[⋆]
[⋆] **Purdue University, ECE Department, West Lafayette, IN 47907**
[†] **HP Labs, Palo Alto, CA 94304**

Abstract

Detecting and aligning structured signals such as point grids plays a fundamental role in many signal processing applications. Joint determination of non-grid points and estimation of non-linear spatial distortions applied to the grid is a key challenge for grid alignment. This paper proposes a candidate solution. The method described herein starts from a small nearly regular region found in the point set and then expands the list of candidate points included in the grid. The proposed method was tested on geometrically transformed point sets and sets of locations derived from imagery of 3D prints. It is shown that a low-complexity grid alignment method can nonetheless achieve high grid alignment accuracy.

Introduction

The identification and interpretation of regular structures or patterns plays a role in many signal processing applications. One structure of interest is a two-dimensional array of regularly spaced locations, a *point grid*. Even when on the surface of an three-dimensional object, for instance, a point grid forms a regular structure. It is straightforward to find instances this structure, i.e, lines present in a small number of points if minimal displacement from the line is allowed [1]. It can be more difficult, however, when the grid is interwoven with a larger number of non-grid points. Applications such as image restoration [2], 3D surface registration [3], and data embedding [4], have utilized point grids. And in these examples, grid alignment, or the mapping of a detected grid to a standard coordinate system, plays an important role.

Many grid alignment methods, representing a special case of point alignment schemes, are discussed in the context of 2D points, as are the algorithms herein. Grid alignment schemes can suffer from both the existence of non-grid points and the presence of nonlinear geometric distortion, and many approaches separate the two obstacles and focus on one. Based on prior knowledge, rule-based approaches [2, 5, 6] can analyze the geometric distortion and then simply align the grid after distortion correction. One solution using CNNs [7] does show that prior knowledge of the distortion is not necessarily required. Nevertheless, these techniques assume the presence of relatively few non-grid points. In fact, non-grid points often exist in acquired signals: they can occur due to noise effects or may be present by intent, such as when the grid serves as a fiducial pattern [4]. An alternative approach is to find the regular grid from a mixed point set. The method in [4] proposed a robust regular-grid alignment method but it does not scale well with grid size, and can become unstable even under distortions due to image perspective.

Joint determination of non-grid points and estimation of non-linear distortions affecting the grid can be a challenge: distortion must be estimated using the grid points but grid points cannot be reliably distinguished from the non-grid points if the distortion is unknown, and vice versa. A piece-wise planar distortion estimation scheme [4] via breadth-first search (PP-BFS) was used to solve this joint problem, which breaks the image into overlapping sub-images. Based on an initial regular grid region found in one sub-image, the distortion is estimated and points within it are classified. The process repeats for adjacent sub-images until all the grid points present are aligned. This method, however, can require tuning to select a sub-image size for robust alignment, and further, can suffer when aligning edge grid points. The proposed approach provides a more adaptive and precise alignment strategy, able to interpret grid signals under a wider range of conditions, such as non-linear transformations or non-contiguous sets of candidate points.

This paper is organized as follows. First, Section 2 formally introduces the grid alignment problem and key assumptions. Then, Section 3 outlines a solution, which implements *atomic* (instead of piece-wise) growth of the candidate grid. Section 4 discusses the performance in different use cases.

Grid Alignment Problem

Figure 1a is an example 2D point set $P = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$, and is a typical input for the grid alignment problem. When solved, grid points and non-grid points can be classified correctly; Figure 1b shows respective examples, as red and blue dots. Point sets such as these may represent SIFT-like key-points [8] extracted from an image, such as is the case here. Our purpose is to find a subset P_G from the entire point set P, such that P_G contains as many grid points as possible, which is the process of determining grid points. We assume that P_G only contains a single grid. We also must determine a mapping g between the grid points and standard (integer, rectified) coordinate system. In other words, we assign a standard coordinate $\mathbf{m}_i$ to each $\mathbf{x}_i \in P_G$ such that $g[\mathbf{m}_i] = \mathbf{x}_i$. The standard coordinate system describes the topological relationship between grid points, i.e., $g[\mathbf{m}_i]$ and $g[\mathbf{m}_i + (1,0)]$ are two adjacent grid points along one direction and $g[\mathbf{m}_i]$ and $g[\mathbf{m}_i + (0,1)]$ are two adjacent grid points along another direction. This also helps to estimate the non-linear geometric distortion. In this paper, we define the k-distance neighborhood of a grid point $\mathbf{m}_i$ to be all the grid points $\mathbf{m}_i + \mathbf{n}_i$ such that $||\mathbf{n}_i||_1 \leq k$.

We assume that most of the grid points are arranged subject to two local constraints, even if the grid is geometrically distorted. In particular, if $\mathbf{x}_{i-1}$, $\mathbf{x}_i$, and $\mathbf{x}_{i+1}$ represent three neighboring candidate grid points, the first constraint indicates a similar spacing, i.e.,

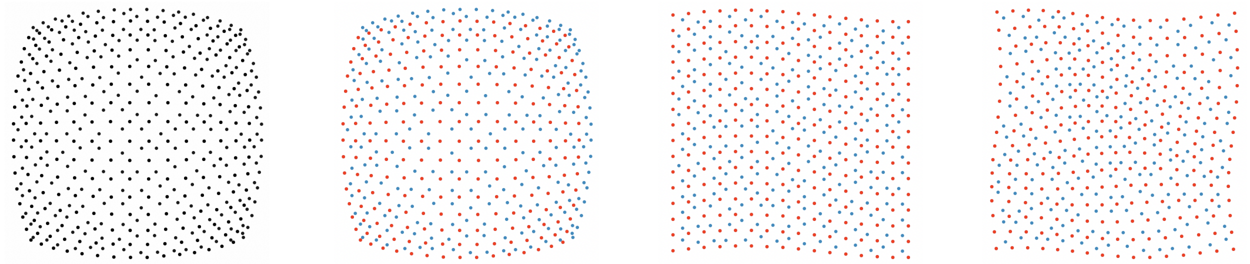
$$\big| ||\mathbf{x}_{i-1} - \mathbf{x}_i||_2 - ||\mathbf{x}_i - \mathbf{x}_{i+1}||_2 \big| \leq \varepsilon_1. \tag{1}$$

The second constraint specifies that the triangle formed by the three neighboring grid points has a largest angle θ such that

$$\theta \geq \pi - \varepsilon_2. \tag{2}$$

The first constraint describes the locally regular spacing; and the second one describes the local orientation along a line. In fact,

*Research supported by HP Inc., Palo Alto CA. U.S.A.



(a) sample perturbed grid, interspersed with spurious points

(b) spherical perturbation, after alignment/classification

(c) sinusoidal perturbation

(d) quasi-random perturbation

Figure 1: Examples of transformed/perturbed point sets, where 1a and 1b compare the points before and after alignment (and then classification) with the proposed method; grid points are presented as red dots, and non-grid points are shown in blue.

one can employ a more conservative constraint. Let

$$\mathbf{x}'_{i+1} = 2\mathbf{x}_i - \mathbf{x}_{i-1}, \qquad (3)$$

and choose a sufficiently small ε. If $||\mathbf{x}_{i+1} - \mathbf{x}'_{i+1}||_2 \leq \varepsilon$, then (1) and (2) both hold. Those constraints often conveniently describe sets of three adjacent grid points along a given direction. Equation (3) gives a method to estimate $\mathbf{x}_{i+1}$ from $\mathbf{x}_{i-1}$ and $\mathbf{x}_i$ that is key to our method.

Atomic Growing

A grid alignment algorithm that features *atomic growth* of the candidate list of aligned points is described below. In essence, it expands an initial grid a point at a time, based on the inferred local geometric structure. The initial regular grid is found using the method in [4] from the given point set P. The initial grid can be small, i.e., four grid points forming the corners of a grid cell are sufficient. In addition, the standard coordinates are derived using the same conventions as in [4].

Algorithm 1: Atomic Growing

```
Result: Map $g$
Determine a set of initial grid points and associated
  image and grid coordinates.
Initialize a map $g$ to relate the grid coordinate $\mathbf{m}$ to the
  image coordinate $\mathbf{x}$ of each grid point.
Create a queue $Q$ and a hash set $S$.
Push all the grid coordinates in $g$ into $Q$.
while $Q$ is not empty do
    $\mathbf{m} = Q.pop()$
    if $\mathbf{m}$ is not in $S$ then
        Estimate $\hat{\mathbf{x}}$ associated with $\mathbf{m}$.
        Find an $\mathbf{x}$ from $M$ which minimizes $||\mathbf{x} - \hat{\mathbf{x}}||_2$.
        if $||\mathbf{x} - \hat{\mathbf{x}}||_2 \leq T$ then
            Set $g[\mathbf{m}] = \mathbf{x}$
            $S.add(\mathbf{m})$
            $Q.push$(grid coordinates of $\mathbf{m}$'s k-distance
              neighbors)
            Remove $\mathbf{x}$ from $M$
```

The step to estimate the original coordinate $\mathbf{x}_i$ associated with a standard coordinate $\mathbf{m}_i$ may depend on the local availability of known aligned and candidate points. One method uses four pairs of adjacent grid points of $\mathbf{m}_i$: one pair immediately above, to the left, to the right, and below $\mathbf{m}_i$. For each pair, we denote the closest point by $\mathbf{x}_{i+1}$ and the second one by $\mathbf{x}_{i+2}$. If any of the two points are not defined in g, this pair is not valid. Each valid pair produces an estimated coordinate $\hat{\mathbf{x}}_i = 2\mathbf{x}_{i+1} - \mathbf{x}_{i+2}$ and an estimated local spacing $||\mathbf{x}_{i+1} - \mathbf{x}_{i+2}||_2$. We may obtain up to four different estimates for $\hat{\mathbf{x}}_i$ (based on the neighboring points) and simply average them to compute a final value. This approach minimizes the effect of non-linear geometric distortion since it estimates the coordinate based on extremely local information. This is the first method. A second method uses a k-distance neighborhood of $\mathbf{m}_i$ that is defined in the dictionary g. A homography H from the standard coordinate system to the original system is calculated based on those points. Next, we project the standard coordinate $\mathbf{m}_i$ to its estimated original coordinate $\hat{\mathbf{x}}_i$ with this homography. Then, we calculate the local grid spacing as the average of $||\hat{\mathbf{x}}_i - g[\mathbf{m}_i + \Delta\mathbf{m}]||_2$, where $\Delta\mathbf{m} \in \{(0,-1),(-1,0),(0,1),(1,0)\}$. If any of those points are not defined via g, we use the homography H to estimate an original associated coordinate instead.

The first method is the primary method in our algorithm. If it fails, the second method is used. Every time a new grid point is determined, all of its k-distance neighborhoods become candidates, ensuring that this grid point has another chance to be determined once the information of its surrounding grid points is updated. This option helps the algorithm to bypass gaps (missing rows and/or columns) resulting in non-contiguous sets of valid grid points. On the other hand, it is also more likely to introduce a larger error to $\hat{\mathbf{x}}_i$ as k increases, due to the presence of non-linear geometric distortion. The threshold T is adaptive instead of being fixed. It is set to the local spacing described above, scaled by a factor ε_s, and thus adjusts with each candidate point.

Results and Discussion

The proposed algorithm was evaluated using both point sets generated under geometric transformations and point sets extracted from real images of 3D printed parts. In addition to evaluating point alignment performance, we employ a form of object marking to test it further. Various object tagging scheme exist [9, 10, 11, 12], and we selected a flexible approach based on point positioning: the grid is used as fiducial markers, and the non-grid point positions encode data as in [13, 14]. Thus, the data decoding accuracy reflects the alignment performance. We use this system to generate grid and non-grid points for both types of tests, and chose alignment parameters suitable for most use-cases, i.e., $\varepsilon_s = \frac{1}{8}$, $k = 7$.

The data embedding system operates as follows. A set of non-grid points are interspersed between points in a regular grid, with each non-grid point representing several bits, present inside each square delimited by the four corner points of every grid *cell*. The decoding process to extract these bits relies heavily on the grid alignment algorithm for two reasons. First, the grid alignment classifies grid points and non-grid points so that the data-bearing points are determined. Second, the grid alignment can correct any distortion introduced to the point set, i.e., print-

	spherical	sinusoidal	quasi-random
mean distance	0.0078	0.0027	0.0030
std. dev. distance	0.0109	0.0013	0.0014

Table 1: Distances statistics between real and predicted (non-grid) data point positions in the standard coordinate system.

ing distortion, camera lens distortion, surface distortion and so on. After the distortion correction, the non-grid points can be located and interpreted as data bits.

Geometrically transformed point sets

The point set P containing a regular grid is generated by the data embedding system. Although it contains non-grid points, we still induce further non-linear distortion by applying a transformation simulating surface rendering. Instead of adding white noise perturbation, we apply smooth perturbation in order to mimic the distortion of applying the grid to a non-flat surface. Figure 1 demonstrates three kinds of perturbation: spherical, sinusoidal, and quasi-random. The quasi-random variant is formed from a generated random perturbation smoothed by a low pass filter. Five different levels of distortion were induced using each type of perturbation.

Let P_G represent the ideal grid and $\hat{P}_G$ our estimate. Correctly aligned grid points, missing grid points and falsely aligned grid points, respectively, are given by $P_G \cap \hat{P}_G$, $P_G - P_G \cap \hat{P}_G$, and $\hat{P}_G - P_G \cap \hat{P}_G$. The performance was measured by the *alignment match rate* and *misalignment rate*,

$$\textit{alignment match rate} = \frac{|P_G \cap \hat{P}_G|}{|P_G|} \times 100\%, \qquad (4)$$

$$\textit{misalignment rate} = \frac{|\hat{P}_G - P_G \cap \hat{P}_G|}{|\hat{P}_G|} \times 100\% \qquad (5)$$

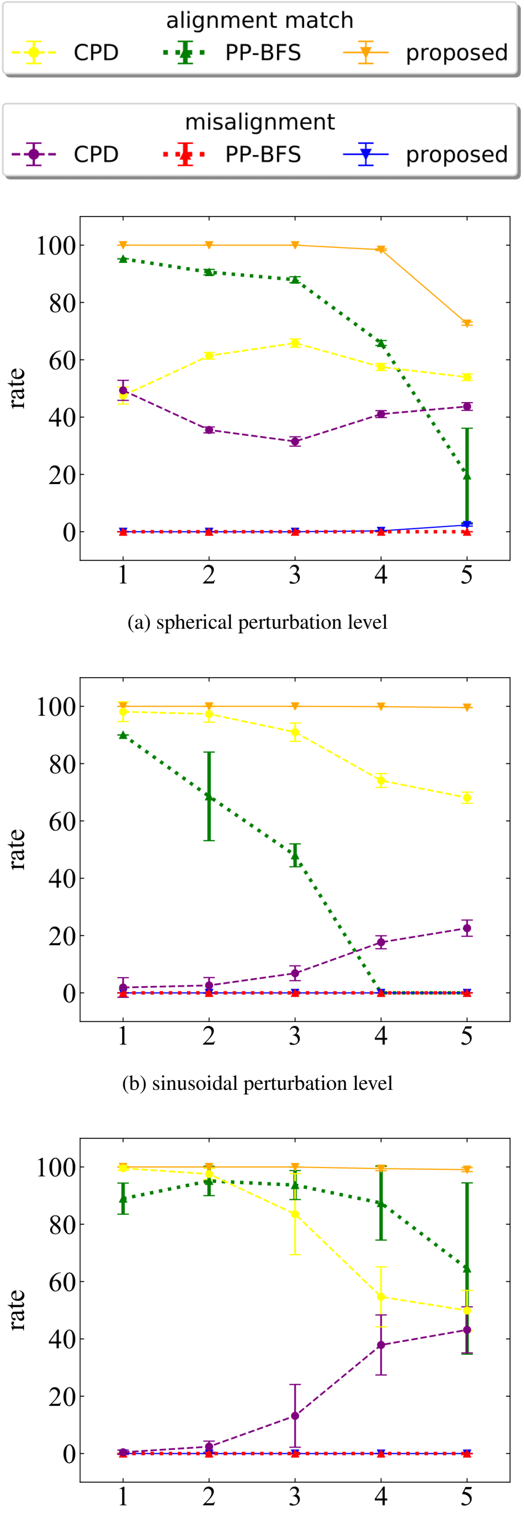
where denotes $|\cdot|$ the set cardinality. Figures 1b, 1c, and 1d feature examples of grid points $\hat{P}_G$ aligned using our method.

The proposed method is compared to two other methods: PP-BFS [4], specifically designed for grid alignment, and a more general point set registration scheme CPD [15] (adapted for our problem). We generated 10 point sets with different messages, i.e., non-grid points, for each kind of perturbation and each perturbation level, 150 point sets in total, to compare the three algorithms. Each point set consists of 29 × 29 grid cells, 900 grid points in total.

The results are displayed in Figure 2. The proposed method yields strong, consistent performance in comparison to the other two approaches, even if extra information is provided, such as the tuned sub-image size for PP-BFS and the (exact) grid size for the CPD method. In tests with sinusoidal perturbation level above four, the alignment matching rate of PP-BFS drops to 0% because it fails to find the initial regular grid points in all sub-images. But atomic growing can endure such perturbation since the initial nearly regular grid points can be found from the entire point set. For the other perturbations, PP-BFS yields better performance than the CPD method. Every technique demonstrated good alignment for low levels of distortion. For the proposed approach, Table 1 suggests spherical distortions presented the most difficult test as the variation in non-grid position estimates was an order of magnitude higher, though *all* messages were successfully decoded under each perturbation.

Features from images of surfaces

We designed parts with controlled patterns of surface features, embossed bumps, to test the ability interpret grid-like arrangements of surface structures. Fiducial grids composed of these features are interspersed between non-grid features representing data as in previous tests, where the positions of the bumps

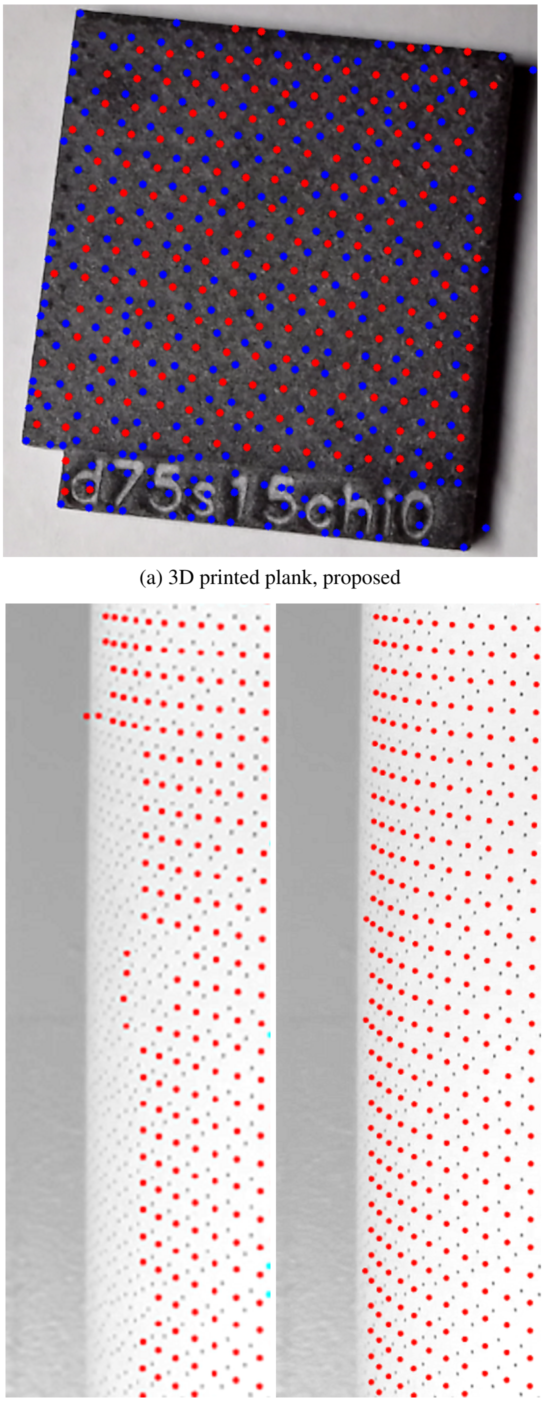


(a) spherical perturbation level

(b) sinusoidal perturbation level

(c) quasi-random perturbation level

Figure 2: Comparison between the proposed approach, PP-BFS [4] and CPD [15] methods applied to geometrically perturbed points. With severe perturbation, the alignment rate of the proposed method is still close to 100%. Note also that the consistence of the proposed approach is superior in most tests.



(a) 3D printed plank, proposed

(b) 2D print on cylinder, PP-BFS (left) v. proposed (right)

Figure 3: Example test objects with red dots representing detected grid points, aligned on a 3D printed part 3a by the method herein, with blue dots indicating non-grid data points, and on a label applied to a cylinder 3b.

are determined using the fiducials. Ten designs were rendered and 3D printed, and test images were captured with an IPEVO Ziggi-HD Plus camera. The point sets were extracted as feature positions via a modified SIFT algorithm; the grid points were subsequently aligned, and the data extracted. The printing and capture process resulted in grid structures that suffered mainly from projective (capture) distortion and pure random (printing) perturbations. Since the test parts were nearly flat, surface irregularities only resulted in minor yet noticeable distortion. A similar set of tests were applied to 2D printed labels on cylindrical surfaces to isolate alignment behaviors from such 3D printing artifacts.

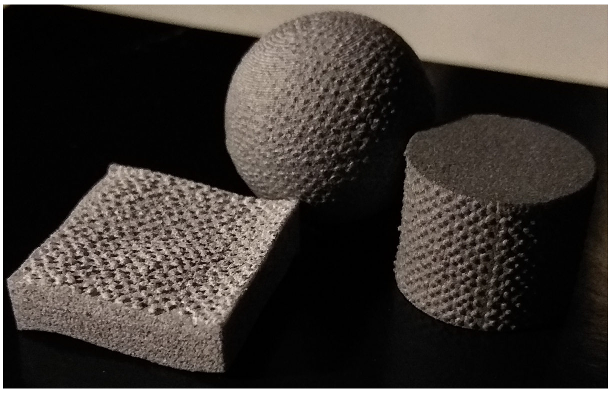
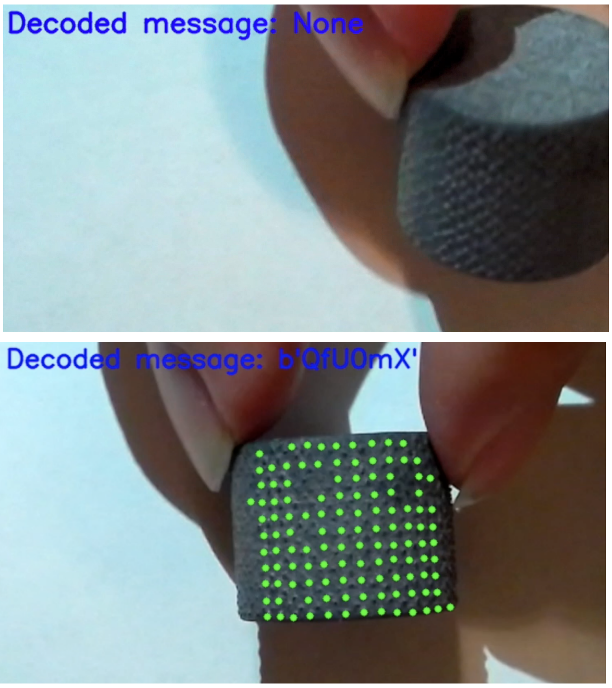


Figure 4: Example non-flat 3D printed test objects that motivate the distortions tested in this paper. Cylindrical, spherical and quasi-randomly perturbed parts surfaces are created to stress the proposed grid alignment technique in different ways, i.e., using strong curvature, 2D non-affine warping, and non-monotonic spatial distortion, respectively.

The proposed algorithm successfully decoded all ten 3D printed planks (zero decoding errors). Despite this result, point sets extracted from real images of printed parts contained a greater number of non-grid points than in the previous test based on geometrically transformed point sets. In Figure 3a, for instance, there are several spurious points on the example plank due to surface irregularities, and many non-grid points on the example label in 3b from the 2D print tests due to contrasting backgrounds and strong edge curvature. In practice, finding 3D-printed features on a surface with the same degree of curvature would be difficult, but testing confirms that alignment can proceed even with it present. On average, the proposed method yielded a 1.1% misalignment rate on the planks, while missing just a few points to achieve a 92.3% alignment matching rate. Most misses were due to feature detection failures near the plank edges(see Figure 3a).

By comparison, the PP-BFS algorithm [4] yielded, on average, a 80.8% alignment matching rate and 0.3% misalignment rate, indicating that this method classifies points as grid points less conservatively than the proposed approach. Figure 3b provides some intuition as to why its alignment matching rate was lower. Due to the need to tune the sub-image size, PP-BFS cannot grow the grid to include many of the points present, and whereas tuning could help in this case, the efficiency does not improve. Similar trends were observed imaging the cylinder at six-, eight- and ten-inch capture distances.

Future work will involve extending such comparisons to use features distributed across more complex surfaces,such as those in Figure 4. Preliminary studies indicate it is possible to extract the same data from aligned grids on *curved* surfaces as has been shown herein to be possible with flat planks. Figure 4 includes a sphere, a cylinder and a wave-like quasi-random generated surface, each with grid-like structures coding data, and in fact, real-time scanning of these objects can be achieved. Figure 5 illustrates frames from a dynamic live video feed processed by the proposed approach: as the cylinder comes into frame and autofocus adjusts to the object, surfaces features are detected, the grid is aligned, and the underlying message is decoded.



Decoded message: None

Decoded message: b'QfU0mX'

Figure 5: Video frames depicting real-time alignment and decoding of patterns on a cylindrical surface. As the surface features move into view and the imaging device adjusts focus, the proposed grid alignment scheme enables correct detection and interpretation of the surface features, despite the fact that they appear as a part of a temporally and spatially varying low-contrast signal.

References

[1] David G. Kendall and Wilfrid S. Kendall. Alignments in two-dimensional random sets of points. *Advances in Applied Probability*, 12(2):380–424, 1980.

[2] Chao Hu, Max Meng, P. X. Liu, and Xiang Wang. Image distortion correction for wireless capsule endoscope. In *IEEE International Conference on Robotics and Automation*, volume 5, pages 4718–4723, 2004.

[3] Ryo Furukawa, Hiroki Morinaga, Yoji Sanomura, Shinji Tanaka, Shigeto Yoshida, and Hiroshi Kawasaki. Shape acquisition and registration for 3d endoscope based on grid pattern projection. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 399–415, Cham, 2016. Springer International Publishing.

[4] Ziyi Zhao, Yujian Xu, Robert Ulichney, Matthew Gaubatz, Stephen Pollard, and Jan P. Allebach. Data-bearing halftone image alignment and assessment on 3d surface. In *Color Imaging XXV: Displaying, Processing, Hardcopy, and Applications*, (Part of IS&T Electronic Imaging 2020), Burlingame, CA, Jan 2020.

[5] Christian Bräuer-Burchardt. A simple new method for precise lens distortion correction of low cost camera systems. In Carl Edward Rasmussen, Heinrich H. Bülthoff, Bernhard Schölkopf, and Martin A. Giese, editors, *Pattern Recognition*, pages 570–577, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[6] Guiyin Hu, Yonglong Luo, Xintao Ding, Liangmin Guo, Biao Jie, Xiaoyao Zheng, and Guorong Cai. Alignment of grid points. *Optik*, 131:279 – 286, 2017.

[7] Dohyun Kim, Dohyun Kwon, Soohyunand Park, and Joongheon Kim. Learning-based dot-grid alignment for projection distortion correction. In *2020 International Conference on Information Networking (ICOIN)*, pages 4–6, 2020.

[8] David Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157, Kerkyra, Greece, 1999.

[9] Dean Fresonke. In-fab identification of silicon wafers with clean, laser marked barcodes. In *Proceedings of 1994 IEEE/SEMI Advanced Semiconductor Manufacturing Conference and Workshop (ASMC)*, pages 157–160, 1994.

[10] Chantrice Mos, Suman Chakrabarti, and David Scott. Parts quality management: Direct part marking of data matrix symbol for mission assurance. In *2013 IEEE Aerospace Conference*, pages 1–12, 2013.

[11] Robert Ulichney and Matthew Gaubatz. Barcodes on non-flat surfaces. In *NIP & Digital Fabrication Conference*, San Francisco, CA, Sep. 2019.

[12] Hao Peng, Peiqing Liu, Lin Lu, Andrei Sharf, Lin Liu, Dani Lischinski, and Baodquan Chen. Fabricable unobtrusive 3d QR codes with directional light. *Computer Graphics Forum*, 39(5):15–27, 2020.

[13] Robert Ulichney, Matthew Gaubatz, and Steven Simske. Circular coding with interleaving phase. In *DocEng 2014*, ACM Symposium on Document Engineering, Fort Collins, CO, Sep. 2014.

[14] Robert Ulichney, Matthew Gaubatz, and Steven Simske. Circular coding for data embedding. In *IS&T NIP29 (29th Int. Conf. on Digital Printing Technologies)*, Seattle, WA, Sep. 2013.

[15] Andriy Myronenko and Xubo Song. Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2262–2275, 2010.

Author Biography

Yujian Xu received his B.S. in mechanical engineering from Shanghai Jiao Tong University. He is a Ph.D. candidate in the department of electrical and computer engineering at Purdue now. His work focuses on detection and alignment of 3D surface properties.