BORDER: An Oriented Rectangles Approach to Texture-less Object Recognition

Jacob Chan[1], Jimmy Addison Lee[2] and Qian Kemao[1]

[1]School of Computer Engineering (SCE), Nanyang Technological University
Block N4 Nanyang Avenue, Singapore 639798
jchan015@ntu.edu.sg, MKMQian@ntu.edu.sg
[2]Institute for Infocomm Research (I2R), Agency for Science, Technology and Research (A*STAR)
1 Fusionopolis Way, Connexis (South Tower), Singapore 138632
jalee@i2r.a-star.edu.sg

Abstract

This paper presents an algorithm coined BORDER (Bounding Oriented-Rectangle Descriptors for Enclosed Regions) for texture-less object recognition. By fusing a regional object encompassment concept with descriptor-based pipelines, we extend local-patches into scalable object-sized oriented rectangles for optimal object information encapsulation with minimal outliers. We correspondingly introduce a modified line-segment detection technique termed Linelets to stabilize keypoint repeatability in homogenous conditions. In addition, a unique sampling technique facilitates the incorporation of robust angle primitives to produce discriminative rotation-invariant descriptors. BORDER's high competence in object recognition particularly excels in homogenous conditions obtaining superior detection rates in the presence of high-clutter, occlusion and scale-rotation changes when compared with modern state-of-the-art texture-less object detectors such as BOLD and LINE2D on public texture-less object databases.

1. Introduction

Recognition of texture-less objects has become increasingly significant in modern times as the world diversifies into 2D-3D research areas such as Augmented Reality and 3D reconstruction/printing. Yet, these objects make detection challenging even for state-of-the-art descriptor-based detectors like the popular SIFT (*Scale Invariant Feature Transform*) [1], and SURF (*Speeded Up Robust Features*) [2]. The key factor that inhibits decent performance in homogeneity originates from the scarcity of local salient information, which impedes the effectiveness of keypoint registration and ultimately degrading the performance of these local patch-based description-matching paradigms. Consequently, this led to many variants of texture-less detection schemes, therefore potentially excluding many descriptor-based virtues such as scale-rotation invariance, model scalability, and the high distinctiveness in the presence of occlusion and clutter.

Current contemporary texture-less object detectors mainly fall into two categories to cope with the lackluster information that these objects resonate. The first technique involves edge/gradient-based template matching [3–9], where objects are trained with various indifferent methods and windowed through the scene to find the best matched location. The popularity of this approach stems from its ability to encompass the object in its entirety, thus granting optimal object description in both textured and homogenous domains at efficient runtimes. However, its robustness quickly diminishes in occluding circumstances, and would need sophisticated amounts of training data to uphold invariances such as rotation, scale and various vantage viewpoints. The second technique assumes an edge-feature aggregation approach [10–14], often adopting a partial SIFT-like pipeline to describe its grouped edge-features. These algorithms typically form interest-points by engaging a line-based representation of the edges, while exploiting various methods to aggregate and describe their associative properties. Despite incorporating virtues like scale-rotation invariance as well as model scalability, this technique often suffers from stability issues especially during occlusion where edges become altered, affecting its size and potentially causing shifts in the interest-points. Furthermore, its spatial feature-grouping approach often degrades due to nearby clutter, corrupting the aggregations and eventually their descriptors.

In this paper, we aim to design a detector that combines texture-less based techniques with qualities from the descriptor-based pipeline to robustly recognize homogenous objects in high clutter and occlusion. Our proposed work BORDER, commences its SIFT-like pipeline with a detection scheme to meaningfully divide elongated line-segments into smaller equal-sized

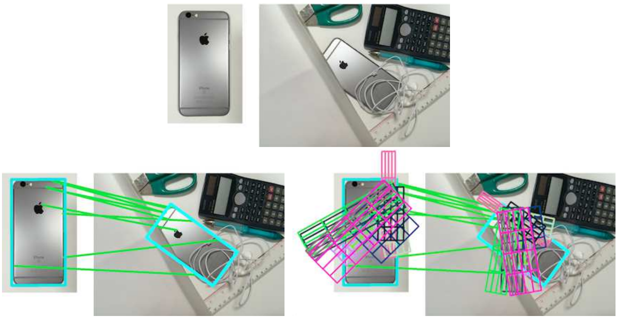


Fig. 1: BORDER's texture-less object detection under clutter and occlusion. (Top) input images, (Bottom-left) feature matching and detection, (Bottom-right) multi-sized oriented rectangle matches.

fragments (termed Linelets) to stabilize interest-point shifts in occlusion. Next, we capture regional object information using an encapsulation concept by acclimating descriptor-based local patches into larger, object-sized scalable oriented rectangles. These rectangles undergo a unique rotational search technique to find ideal positions to optimally describe the object from a keypoint's "point-of-view". Furthermore, we deploy BORDER in a multi-sized rectangle scheme to incorporate sub-sectional encapsulation for added occlusion resistivity. Subsequently, encompassed regions undertake a linear sampling process to accumulate state-of-the-art rotation-invariant angle primitives [10] to form its descriptors. Finally, we match BORDER descriptors by exercising the randomized kd-tree forest [15] after a pre-processing procedure. Fig. 1 presents an object detection instance of BORDER with its multi-sized oriented rectangle scheme in a cluttered scene.

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3 presents the BORDER methodology. Section 4 showcases the experiment results. Lastly, section 5 concludes this work.

2. Related Work

Given its impressive detection rates even in the presence of clutter and occlusion, BOLD (Bunch of Line Descriptors) [10] is arguably the current leading texture-less descriptor-based detector. It uses detected line-segment midpoints as interest-points to aggregate nearby segments via k nearest neighbors (kNN), while describing them using their unique angle primitives. However, as revealed in their paper, nearby distractors tend to cause aggregations to include unwanted segments. Furthermore, line-segment midpoints may be susceptible to shifts due to various factors making the repeatability of the interest-points questionable. Another work that resembles BOLD is [11], where lines are similarly employed for detection, but are described by a pairwise structure. Geometrical, color and distance relationships within each line pair then proceed to train its classifier. Other works of similar nature such as [12, 13] likewise use line aggregations, though with indifferent primitives and training methods. Damen et al. [14] proposed using edgelets (short straight segments defined by its midpoint and orientation) to accumulate traced angled paths by reflecting off other edgelets to form constellations. The resultant collection of path orientations and distances are subsequently encoded to form its descriptors. Even though this method feels similar to BOLD, it lacks descriptor distinctiveness. Moreover, its path reflection method is very sensitive to occlusion and noise, as constellation paths would differ due to missing or additional edgelets. Other pioneering algorithms of significance in this genre includes a triple-edge feature with a point-based geometric hashing comparison method by Proctor and Illingworth [16], and a cubist approach by Nelson and Selinger [17] using boundary segments called key curves to generate patches for feature extraction.

Template-based matchers are another widely researched area of texture-less object detection. One of the most significant work termed LINE2D (2D templates) or LINEMOD (2D templates+depth maps) [3, 4] uses quantized gradient orientations to form response maps for input windowing comparisons through a similarity measure. LINE2D achieved real-time detection while producing decent results in homogeneity, but quickly degrades under clutter and occlusion. Hsiao and Herbert [18] followed up by attaching an occlusion reasoning component to LINE2D, enhancing its detection rates under highly distractive scenes. Even so, LINE2D's template-based origin leads to scalability issues, often requiring voluminous number of templates for recognition in variances such as rotation, scale and perspective changes. Other notable template-based works include techniques such as chamfer matching [6, 7, 19] and Hausdorff distance [8, 9], which utilize edge-point templates to find correspondences using a similarity measure. Additionally, 3D template-based algorithms using 3D CAD models [20] and depth map templates [21] were respectively introduced in recent years due to the availability of auxiliary depth information.

Finally, shaped-based learning methods also made substantial impact in texture-less detection. Ferrari et al. [22, 23] introduced a local contour feature called k-Adjacent Segments (kAS) by linking edgel-chains (fairly straight contour segments) and describing them using their orientations and lengths. These kAS features consolidate into class-specific codebooks to learn the object's shape model for detection using a Hough voting scheme. Other contour-based shape detectors comprise of [24–26], where regions are similarly detected using contours and learned using shape context descriptors. Although these works displayed decent performance in clutter and scale, rotation and/or occluding results were largely unreported.

3. The BORDER Methodology

BORDER's recognition scheme follows the 3-step SIFT-like pipeline of detection, description and matching. This is accompanied by a regional object encompassment concept by first detecting interest-points via linelets, followed by a description scheme focused about oriented rectangles. We use linelets as pivot anchor-points for oriented rectangles to discover good locations to describe the object, and subsequently match them based on a unique "point-of-view" scheme.

3.1. Linelet Detection

Line-segments have shown in modern times to effectively present a low-level stable edge feature representation for texture-less objects [10, 13, 14]. The Line Segment Detector (LSD) developed by Von Gioi et al. [27], enables images to be expressed in terms of lines with minimal need for manual parameter tweaks. However, when purely used as a keypoint detector, it encounters a major complication particularly with elongated lines. These lines often materialize in low-curvature areas, thus facilitating region-growth through their related neighboring orientated pixels. Consequently, as line-feature based detectors tend to exploit segment-centers as interest-points [10–14], any occluding alterations of these lines will render stability issues due to midpoint shifts. Hence, to counteract this shortcoming, we propose an adaptation of LSD to generate equal fragmented versions of extensive line-segments termed Linelets.

The underlying principle behind linelets is to intuitively fragment stable LSD-produced line-segments based on a model-scene proportion concept. This is done as opposed to sampling edges at regular intervals [11, 14], to reduce redundancy and meaningfully tune fragment-width according to the severity of clutter and occlusion. We begin by applying an initial LSD detection step for both input images to obtain the line segments of the model $\mathbb{L}_{model}$ and the scene $\mathbb{L}_{scene}$ respectively. Subsequently, we initiate the modification by defining a width threshold

$$\omega = \tau \mathcal{R}_{min}, \tag{1}$$

where $\tau \geq 1$ refers to the width-limit factor with $\tau = 1$ denoting that fragment widths are restricted to $\mathcal{R}_{min}$. The variable $\mathcal{R}_{min}$ indicates the minimum region size for line-segment validation established through the NFA (Number of False Alarms) computation theorized by Desolneux et al. [28], and extensively tested in [29] for automatic detection without tuning intervention for LSD. This parameter is an ideal base parameter for ω since all lines produced from LSD region-growing must contain at least $\mathcal{R}_{min}$ aligned pixels to be deemed as a valid line-segment. Next, to incorporate our proportion-based fragmentation concept, we define

$$\tau = \frac{\mathbb{L}_{max}}{\mathcal{R}_{min}} \cdot \frac{n(\mathbb{L}_{model})}{n(\mathbb{L}_{scene})} \cdot \frac{\overline{|\mathbb{L}_{scene}|}}{\overline{|\mathbb{L}_{model}|}}, \tag{2}$$

where $\mathbb{L}_{max}$ is the model object's longest line-segment, $n(\mathbb{L}_{scene})$ and $n(\mathbb{L}_{model})$ refers to the total model-scene line-segments detected, while $\overline{|\mathbb{L}_{scene}|}$ and $\overline{|\mathbb{L}_{model}|}$ denotes the average lengths of all model-scene line-segments respectively. The first term in Eq. (2), $\mathbb{L}_{max}/\mathcal{R}_{min}$ indicates the maximum fragments producible from the model object's longest line-segment $\mathbb{L}_{max}$ using $\mathcal{R}_{min}$. We chose $\mathbb{L}_{max}$ due to its fragmentation priority, as the longest uninterrupted model line-segment would likely need the most partitions to keep it stable against scene occlusions. The subsequent expressions incorporate our proportion estimates where $n(\mathbb{L}_{model})/n(\mathbb{L}_{scene})$ takes account of the model-scene line-segment density to estimate clutter, and $\overline{|\mathbb{L}_{scene}|}/\overline{|\mathbb{L}_{model}|}$ approximates the line-sizing ratio to verify occlusion breakages using the model-scene average lengths. Lastly, we finalize ω by setting its boundary to

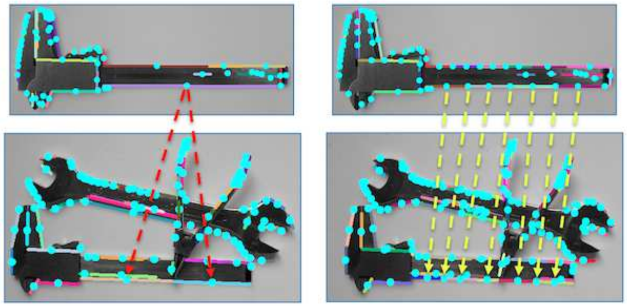


Fig. 2: Stability comparison between line-segment and linelet detection. (Left) LSD detection where a line has been split into two by an overlaid object. (Right) Linelets on the other hand demonstrates better resistance in the presence of occlusion.

$$\omega_\ell = \min[\max(\omega, R_{min}), \mathbb{L}_{max}], \tag{3}$$

where ω_ℓ represents the standardized width of each fragment for both input images, which is used to divide the previously materialized line-segments that have lengths $\geq 2\omega_\ell$ into equal ω_ℓ widths to produce linelets. Note that if $\omega_\ell = \mathbb{L}_{max}$, it implies that linelets fall back to line-segments. This happens when clutter and occlusion are both presumably low, thus requiring no partitions. Finally, each linelet ℓ is attached with properties such as its midpoint m_ℓ, its orientation direction θ_ℓ, and the aligned pixels $a_i \in \mathcal{A}_\ell$ gathered from LSD region-growing. Fig. 2 shows a stability comparison between line-segments and linelets.

3.2. The Oriented Rectangle Template

Before beginning the description process, the initial shape of the oriented rectangle $\mathcal{OR}$ must be pre-determined. This preliminary step aims to attain the ideal dimensions of the $\mathcal{OR}$ to optimally encapsulate the input model object. We surveyed numerous public databases and branded template images into three main categories. The first type is basically an undistracted template of the object, the second involves a scene with the object prominently presented alongside various insignificant distractors, and the last category suffices a scene with an object mask. All three models can be automatically enclosed by the combination of linelet detection and the minimum enclosing box algorithm [30], with the second model requiring an additional salient region detector with automatic threshold [31] to obtain its object mask. Alternatively, BORDER also permits for manual $\mathcal{OR}$ dimensional definition. Note that all models

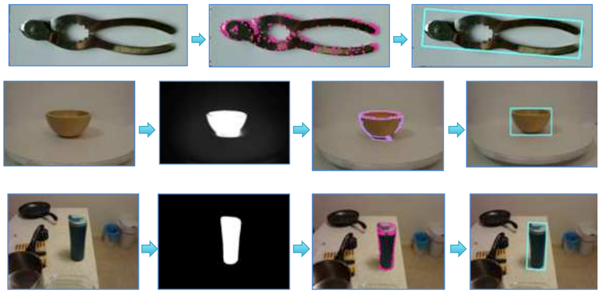


Fig. 3: Automatic $\mathcal{OR}$ prototype detection schemes. (Top-row) Undistracted model template. (Middle-row) Salient model template. (Bottom-row) Scene-mask model template.

in our experiments were encapsulated by the automatic method, as shown in Fig. 3. The $\mathcal{OR}$ prototype serves as the basis for several descriptive purposes. First, it enables length/breadth tuning for generating multi-sized $\mathcal{OR}$s for promoting diverse sub-area object encapsulation to facilitate robustness to occlusion. Next, it also caters for $\mathcal{OR}$ scale factor σ adjustments for finer scale-space variations (e.g. $\sigma = 0.7$) as opposed to image pyramids. Finally, $\mathcal{OR}$s are sub-divided into equal $4{\times}4$ blocks for description purposes. Although other dimensions can be set, we have found that this setting offers the best trade-off between descriptor performance and vector size.

3.3. Linelet-$\mathcal{OR}$ Association and Revolution

Typically, it is customary for a local-patch based detector to center its keypoint on a description region (e.g. SIFT [1] and SURF [2]). However, given the object encapsulation concept of BORDER, linelet midpoints m_ℓ are instead placed at the corner of the $\mathcal{OR}$. This is done with the assumption that linelets would habitually be detected around object edges. Therefore, by aligning the corner of an $\mathcal{OR}$ with a linelet, it could hypothetically promote idealistic encompassment of the object with minimal outliers. For standardization, we align the $\mathcal{OR}$ according to the linelet's direction at its longer side. After association, the $\mathcal{OR}$ proceeds for a scheduled full revolution about m_ℓ. This rotation step plays an important role in BORDER as it allows each m_ℓ to search for good locations to describe the object from its own "point-of-view". The rotation takes place n_r times, angled equally at $\theta = 2\pi/n_r$ per sample r. Moreover, as each placement is essentially a halfspace due to its corner-based association, we correspondingly take account of its flipside to ensure all facets of the revolution equivalently cover the entire spectrum. Consequently, this brings the total rotations to $2n_r$. Fig. 4 demonstrates the $\mathcal{OR}$ revolution about a linelet midpoint after association.

3.4. Descriptor Formulation Criterion

For a rotation sample r to be deemed describable it has to attain an encapsulation standard, otherwise the rotation sample is skipped without having a descriptor being built.

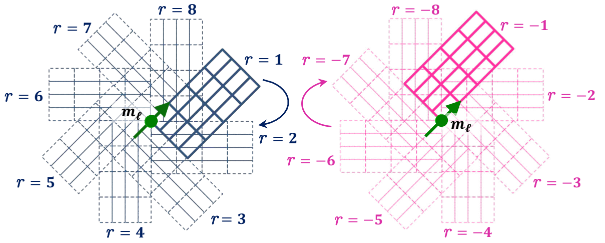


Fig. 4: Revolution sequence ($n_r = 8$) of an associated $\mathcal{OR}$ about a linelet midpoint (left) and its flipside (right).

The criterion governing descriptor formulation is as follows

$$desc_\ell(r) = \left[\frac{n_\ell(r)}{\sigma n(\ell_{model})} \cdot \log_{n_B}[n(v_B(r))] \geq \gamma\right], \quad (4)$$

where $desc_\ell(r)$ returns a boolean regarding the description validity of the r^{th} rotated $\mathcal{OR}$, $n_\ell(r)$ is the total linelets within the r^{th} rotated $\mathcal{OR}$ of the ℓ^{th} linelet, $n(\ell_{model})$ denotes the model's total linelets, σ is the current scale of the $\mathcal{OR}$, n_B refers to number of $\mathcal{OR}$ blocks (e.g. $4 \times 4 = 16$), $n(v_B(r))$ represents the number of valid blocks (blocks with at least one linelet), and γ is the description validity factor. Note that the log-function in Eq. (4) balances the weight of the block validity ratio due to the relatively-low counts of texture-less objects. Also, if $n(v_B(r)) = 0$, or $n_\ell(r) > \sigma n(\ell_{model})$ the rotation sample will be omitted, as the former constitutes an empty $\mathcal{OR}$, while the latter signifies corruption because rotation samples should never contain more linelets than the model. Overall, we apply this authoritative condition to each model/scene image pair to enforce the abundance and distribution of linelets within the $\mathcal{OR}$s for strong descriptor significance. Any rotation sample that meets the threshold is labeled as a prospective descriptor, therefore potentially producing multiple keypoints at the same location with each tagged with a different r. Fig. 5 demonstrates the full descriptor validity process.

Before the online application of the condition, the appropriate γ has to be established. For this, all model linelets undergo a pre-revolution step using the prototype (section 3.2) while applying Eq. (4) to obtain the maximum score γ_{max} for each linelet revolution. Subsequently, we gather all linelets' γ_{max} and assign the minimum value of the collection as the description validity factor γ. This asserts that all model linelets would produce at least one descriptor from its maximum score after applying Eq. (4). After establishing γ, we re-iterate the pre-revolution step with different $\mathcal{OR}$ sizes by reducing one row/column at a time (e.g. $4{\times}3$, $3{\times}4$, …, $2{\times}2$) from the original $\mathcal{OR}$ prototype and re-dividing it to $4{\times}4$ blocks. Any reduced $\mathcal{OR}$'s encapsulation found to surpass γ will also have its length/breadth deployed as part of the multi-sized descriptor scheme within each scale space iteration. This is another key characteristic of BORDER in addition to linelets to greatly enhance detection rates during occlusion.

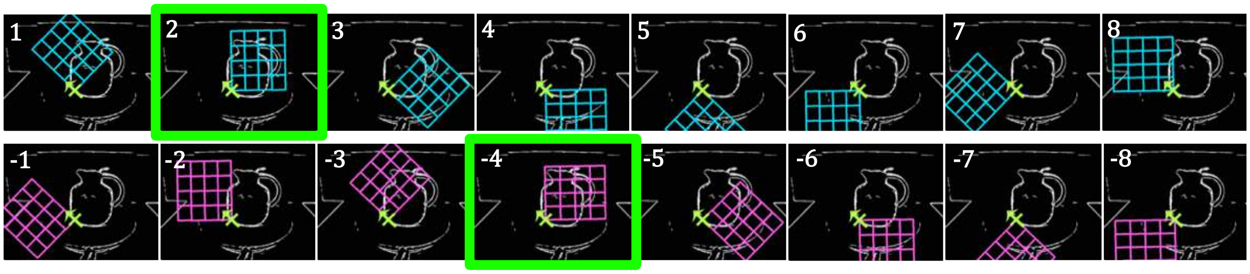


Fig. 5: Rotation sequence of the oriented rectangle (top) and its flipside (bottom) with $n_r = 8$. The rectangle rotates at an incremental $2\pi/n_r$ per sample. In this case, samples 2 and -4 (flipside) will be chosen for description (highlighted in green).

3.5. BORDER Description

For each approved rotational position $desc_\ell(r) = 1$, empty blocks are assigned with zeros in its particular histogram block vector, whereas valid blocks proceed with the description. For each valid block, we start by consolidating the encapsulated linelets' aligned pixels $\mathcal{A}_\ell$ for sampling. This is done as opposed to using every pixel within the block to keep updates to a minimal. Furthermore, as pixels within each $\mathcal{A}_\ell$ have closely related orientations from LSD region-growing, we are able to sample $\mathcal{A}_\ell$ using a stepsize approach without sacrificing performance. Let B be the histogram block scheduled for update and $\mathcal{A}_B = [\mathcal{A}_\ell \in B]$ be the set of aligned pixels within B, then we define the number of update samples n_s for B as

$$n_s = \sigma \gamma \mathcal{R}_{min} n(\ell_B), \tag{5}$$

where σ is the current scale of the $\mathcal{OR}$, $0 \leq \gamma \leq 1$ is the description validity factor (section 3.4) used as the object-complexity indicator to normalize the sample size based on the model object's information richness, $n(\ell_B)$ is the number of linelets in the current B, and $\mathcal{R}_{min}$ refers to the minimum region size validation parameter (section 3.1) used as a standardized $\mathcal{A}_\ell$ size for all linelets. Subsequently, the block's stepsize can be represented by

$$stepsize = max\Big(\big\lfloor n(\mathcal{A}_B)/n_s \big\rfloor, 1\Big), \tag{6}$$

where $n(\mathcal{A}_B)$ refers to the total aligned pixels within the current B. Finally, each sampling pixel is retrieved using

$$s_i = \mathcal{A}_B(stepsize \cdot i), \; 0 \leq i \leq n_s, \tag{7}$$

where s_i refers to i^{th} sampled pixel. In general, this stepsize approach normalizes the sampling rate for each linelet's $\mathcal{A}_\ell$ according to its size within B to minimize updates.

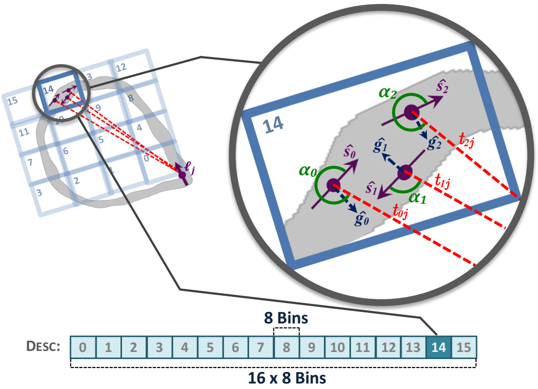


Fig. 6: Three samples and a linelet forming pairs and angle primitives. Each α_i updates its respective block histogram vector. Blocks are concatenated according to its assigned block number.

For the descriptors, the key factor besides delivering exclusiveness is rotation invariance. Out of the existing texture-less description methods [11–14], BOLD's pairwise geometric primitives has proved to be the most robust as presented in their paper [10]. Therefore, to acclimatize our block samples into similar robust angle primitives, we simulate each sample s_i and its origin linelet keypoint ℓ_j as point pairs while extending each s_i into unit vectors using its aligned direction as shown in Fig. 6. This creates two angles, but we only use the primitive at s_i as the angle at ℓ_j has already been encoded by the $\mathcal{OR}$'s rotation index and its block sequencing. Contrary to BOLD, information gathered during linelet detection allows us to simplify the derivation of our single-sided angle primitive α_i to

$$\alpha_i = \begin{cases} \arccos\left(\dfrac{\hat{s_i} \bullet t_{ij}}{\|t_{ij}\|}\right), & \hat{g_i} \bullet t_{ij} < 0 \\ 2\pi - \arccos\left(\dfrac{\hat{s_i} \bullet t_{ij}}{\|t_{ij}\|}\right), & otherwise \end{cases}, \tag{8}$$

where $\bullet$ is the dot product, t_{ij} refers to the imaginary vector between s_i and its origin linelet keypoint ℓ_j, while the unit vectors $\hat{s_i}$ and $\hat{g_i}$ represents the aligned direction and gradient orientation of s_i respectively. The effectiveness of this primitive can be attributed to its contrast polarity property, which is incoporated by verifying the pointing-directions of $\hat{g_i}$ and t_{ij} with respect to its $\hat{s_i}$ axis. For instance, if both $\hat{g_i}$ and t_{ij} point to different halfspaces with $\hat{s_i}$ as their separating axis (e.g. $\hat{s_1}$ in Fig. 6), then we assign the smaller angle between $\hat{s_i}$ and t_{ij} to α_i, otherwise the larger inverse angle is allocated instead (e.g. $\hat{s_0}$ and $\hat{s_2}$ in Fig. 6). By doing so, it enables additional robustness by embedding the direction of s_i in the descriptors. All α_i gathered from each sample-origin pair form votes with weights corresponding to the gradient magnitude of its respective samples. We assign each histogram block with 8 orientation bins ($\theta = 2\pi/8$), while

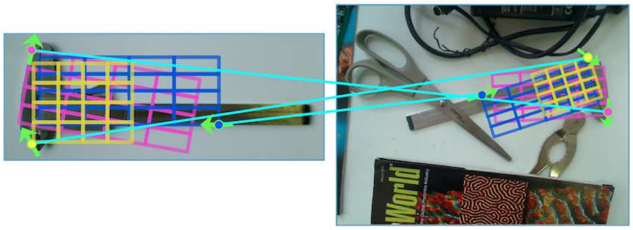


Fig. 7: BORDER deployed with multi-sized $\mathcal{OR}$s matched with similar r-tag rule. The first match (purple) is at $r = 3$, the second match (blue) at $r = 1$, and the third match (yellow) at $r = -6$. Each linelet's direction is indicated by the green arrow.

employing bilateral accumulation to minimize effects of quantization. Finally, all block histograms are concatenated to form a $16 \times 8 = 128$ dimensional vector with the first block sequence being nearest to the origin as indicated in Fig. 6. As for normalization, tests showed that $L2$ norm applied individually block-wise rather than the entire concatenated descriptor offered the best results. This is primary because $\mathcal{OR}$s span over large regions, therefore becoming susceptible to vastly contrasting magnitudes.

3.6. Matching BORDER Descriptors

BORDER asserts that each descriptor can only be matched with another that has been produced at a similar "point-of-view". This means that descriptors are only compared when they are created at the same rotation index r, as illustrated in Fig. 7. We introduce this rule due to the tendency of texture-less objects to be symmetrical, hence increasing its matching ambiguity. To accommodate this condition, we distribute the descriptor vectors into $2n_r$ spaces according to its tagged r in the description phase. Therefore, no additional memory or work is sacrificed for the preparation as opposed to accumulating into a single large space. As for the actual matching phase, only vector spaces that correspond to the same r-tag between the train and query descriptors gets matched. BORDER utilizes the randomized kd-tree forest from FLANN [15] for Euclidean distance matching followed by a geometric verification process for object localization within the scene.

4. Experiments

This section compares BORDER against other contemporary detectors in both texture-less and textured genres. To ensure comprehensiveness, we employ modern texture-less object detectors such as BOLD [10] and LINE2D [3], as well as popular textured-based keypoint detectors like SIFT [1], SURF [2] and ORB [32]. A total of three datasets have been chosen for our experiments, the *D-Textureless dataset* [10] to challenge BOLD's high detection rates, the *CMU Kitchen Occlusion dataset (CMU-KO8)* [18] for its highly cluttered and occlusive scenes, and finally a textured-based assessment using *The Stanford Mobile Visual Search (SMVS) Data Set* [33]. A preview of the datasets can be seen from the feature matching results of BORDER in Fig. 12, 13, and 14 respectively.

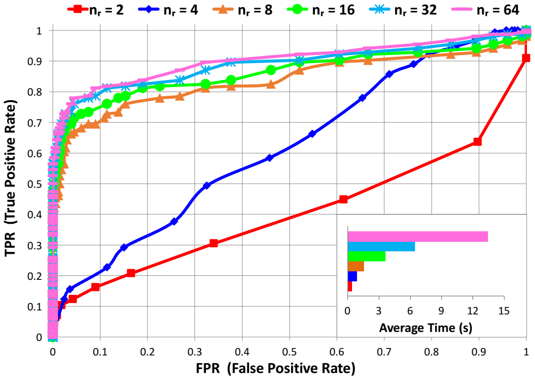
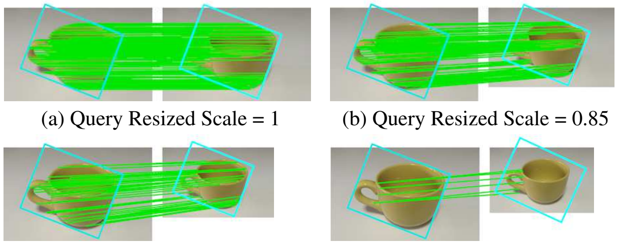


Fig. 8: Experiment for optimal rotation samples n_r.



(a) Query Resized Scale = 1

(b) Query Resized Scale = 0.85

(c) Query Resized Scale = 0.75

(d) Query Resized Scale = 0.7

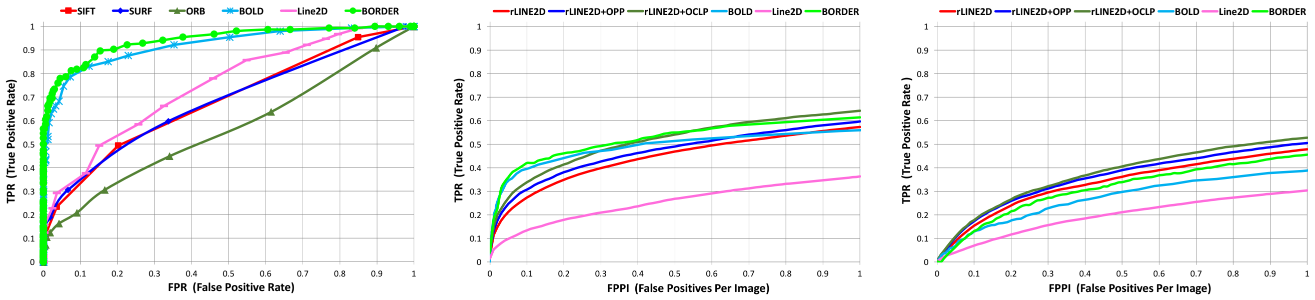
Fig. 9: Example of the scale space experiment. (a), (b) and (c) shows resized versions of the query images with object detection remaining valid. (d) demonstrates that at scale more than 0.25, object detection starts to degrade.

4.1. OR Rotations and Scale Space Intervals

Before the comparisons, internal tests were conducted on BORDER to validate its undetermined parameters.

$\mathcal{OR}$ rotations test This first test involves n_r, the number of rotation samples (section 3.3) needed for each linelet-$\mathcal{OR}$ association. This experiment iterates all the mentioned datasets while increasing n_r in multiples of two to record its detection versus rotation samples trend. From the graph in Fig. 8, it can be observed that greater samples of n_r offer better detection rates at the expense of increased complexity. However, as rotation numbers climb, saturation starts to occur, thus rendering high samples impractical. Therefore, we deduce that $n_r = 8$ offers the best compromise between detection and complexity rates.

Scale space test Mentioned in section 3.2, BORDER exploits the $\mathcal{OR}$'s scale factor σ for scale invariance. Therefore, it is paramount to uncover the scale tolerance of the $\mathcal{OR}$ in order to minimize iterations. For this experiment, we repeat BORDER using the same object for both inputs with the query image resized at 0.05 intervals, while keeping both $\mathcal{OR}$s' $\sigma = 1$. We apply this method to all the dataset models and conclude that keeping σ at intervals of 0.25 (e.g. $\sigma = 1, 0.75, 0.5, 0.25$) provides the best tolerable scale space coverage. Fig. 9 demonstrates the detection



(a) D-Textureless

(b) CMU-KO8 Single-view

(c) CMU-KO8 Multi-view

Fig. 10: Texture-less object database experimental results. (a) D-Textureless Tools dataset, (b) and (c) CMU-KO8 Kitchen Occlusion single-view and multi-view datasets respectively.

degradation as the object is resized. Note that BORDER only scales σ at the query image during actual execution.

4.2. Comparative Results

With BORDER's parameters all tested and automated, we proceed with the experimentations. For the competing algorithms such as LINE2D, SIFT, SURF and ORB, they were all implemented in C++ set with their proposed parameters. BOLD on the other hand was realized by the library from their project site [10]. Apart from ORB, all interest-point detectors including BORDER uses the Euclidean distance-based FLANN-randomized kd-tree for descriptor matching. ORB instead employs the FLANN-LSH matcher as recommended in their paper [32]. To inscribe further fairness in the findings, all keypoint-point based detectors follow the Hough-voting scheme [1] for incremental accumulation of the curves. The processors used for the tests runs at 1.7GHz dual-core from the Intel Core i7 Haswell line, with RAM up to 8GBs.

D-Textureless dataset experiment The first experiment engages the *D-Textureless dataset* by the creators of BOLD. It contains 9 templates of commonly used tools, accompanied by 55 scenes. Besides being texture-less, this dataset challenges algorithms on properties such as translation, rotation, and up to about 50% scale and occlusion. We line up all participating algorithms including LINE2D, as *D-Textureless* provides its training data, to obtain the ROC plot as shown in Fig. 10a. From the results, we observe that the texture-less based detectors clearly outperform the others, with BORDER able to slightly edge out BOLD to claim top spot. Although the result between the two may be marginal, it is very significant upon analysis. As prior to the experiment, we found that BOLD had already achieved an impressive 86% true positive rate at its default settings on *D-Textureless*. Therefore, to be able to surpass BOLD in this dataset, truly exemplifies BORDER's robustness in detecting such texture-less objects. Deeper investigations uncover that BORDER performs better in cases shown in Fig. 12, where objects have their extensive lines occluded. This can be accredited to linelets for its line breakage resistance, together with BORDER's isolative description methods.

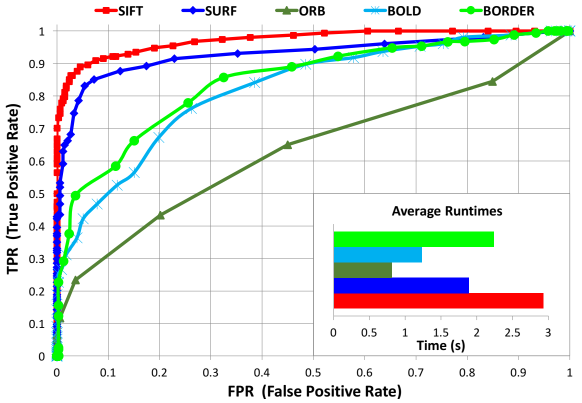
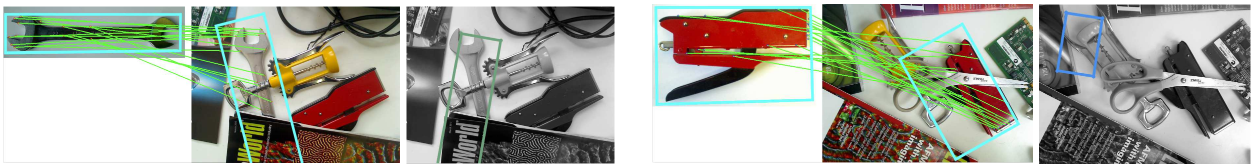


Fig. 11: Experiment results from the textured SMVS dataset.

CMU-KO8 dataset experiment The next experiment involves the extremely cluttered and occluded CMU Kitchen Occlusion dataset (CMU-KO8). Assembled by Hsiao and Herbert, this dataset was built to evaluate their occlusion reasoning model, which was used in conjunction with LINE2D for enhancing performance under highly occlusive scenes. It models 8 texture-less kitchenwares with object masks provided, together with 100 scene images for each object in both single (8x100) and multi-view situations (8x100). The key challenge this dataset presents is the various challenging levels of clutter and occlusion, set in a texture-less domain. Hence, very little emphasis was placed in scale, rotation and even translation. We consolidated the results of BORDER, BOLD, LINE2D and LINE2D+Occlusion reasoning (rLINE2D, rLINE2D+OPP and rLINE2D+OCLP) in a recall vs. FPPI (False Positives Per Image) scheme as portrayed in [18] to compare the average detection results of both the single and multi-view cases. Textured-based detectors are not compared because of the lack of keypoint registration for most of the model objects. Fig. 10b and 10c shows the performance of BORDER over the others in both the single and multi-view databases respectively. In this case, there is a clearer distinction between the detection rates of BORDER and BOLD, with the former having about a 7% lead over



(a) (b) (c) (d)

Fig. 12: Recognition results from D-Textureless where BORDER (a), (c) successfully detects, whereas BOLD (b), (d) falls short.

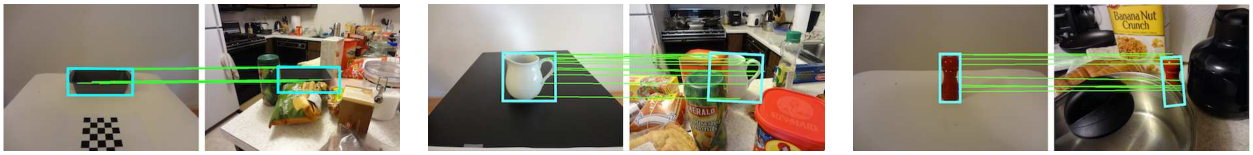
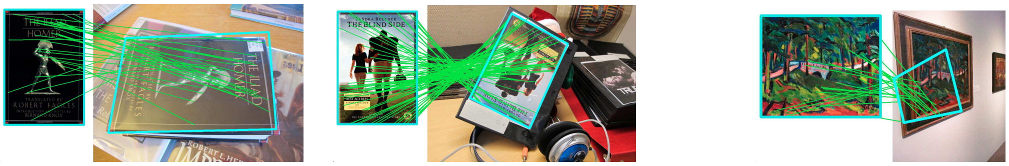


Fig. 13: Some BORDER impressive feature matching results from the CMU-KO8 dataset.



(a) BORDER's positive feature matching results for textured objects.

(b) Failed to detect under high affine change.

Fig. 14: BORDER's SMVS dataset textured sample results. Favorable results from these textured objects can be seen from (a), (b) demonstrates a futile attempt to detect under a high angled viewpoint.

the latter in both datasets. LINE2D's performance was quite mediocre, but addition of the occlusion reasoning models propelled it to a much more competitive level. Overall, BORDER achieved the best detection rates among the texture-less detectors, as well as obtaining almost similar results against the LINE2D+occlusion reasoning models without any reasoning intervention. Therefore, this establishes BORDER's recognition robustness in heavy occlusion as well. Fig. 13 exhibits some remarkable feature matching results of BORDER on *CMU-KO8*. Additionally, the complete BORDER's feature matching results of *D-Textureless* and *CMU-KO8* can be found in our database[1].

SMVS dataset experiment Thus far, with the preeminence of BORDER in both the *D-Textureless* and the *CMU-KO8* datasets, we can safely acknowledge its robustness in detecting texture-less objects. Consequently, to have a complete evaluation in terms of general object detection, we conduct an experiment on a textured database from the *SMVS dataset*. This dataset contains 8 categories such as *book covers, business cards, cd covers, dvd covers, museum paintings, print, landmarks and video frames*. Among the 8 we have left out the last two, as they are not object-based. Each remaining category comprises of 100 models for training, and 400 query images for testing respectively. We choose this dataset primarily due to its diversity as it examines algorithms on different textures such as texts, artworks as well as posterized images. Fig. 11 reports the average ROC results of the detectors used in this dataset, while Fig. 14 shows some BORDER feature matching results. Note that LINE2D is left out due to insufficient templates for its training. As anticipated from the results, SIFT/SURF completely dominates, while BORDER on the other hand outperforms BOLD. One aspect that texture-less detectors falls short is in high perspective viewpoints situations. This is due to their large region-based detection schemes, which suffers severe degradation in affine changes as seen in Fig. 14b.

Although achieving state-of-the-art detection rates, BORDER's high rotations/samples per linelet ultimately costs its runtime to be the higher than BOLD as shown in Fig. 11. It is however quicker than SIFT, and would be multitudes faster after parallel/GPU intervention.

5. Conclusion

We have presented a detector termed BORDER, which combines a regional object encompassment concept with descriptor-based pipelines to recognize texture-less objects in the presence of high clutter and occlusion. The algorithm stabilizes interest-points in the form of linelets and delivers effective descriptor formation with its oriented-rectangle revolution scheme. BORDER is also invariant to scale and rotation which is vital in today's real-world applications. Results from three datasets revealed BORDER's superior recognition rates among the state-of-the-art texture-less detectors, while displaying high competence in textured instances as well.

[1] https://www.dropbox.com/sh/87trs7j798ottbq/AADY4bgAor9G5IOAzHcXCTQ0a?dl=0

References

[1] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 1, 4, 6, 7

[2] H. Bay, T. Tuytelaars, and L. J. Van Gool. Surf: speeded up robust features. In *Proc. ECCV*, volume 3951, pages 404–417, 2006. 1, 4, 6

[3] S. Hinterstoisser, C. Cagniart, P S. Ilic, N. Navab Sturm, P. Fua, , and V. Lepetit. Gradient response maps for real-time detection of textureless objects. *PAMI*, 34(5):876–888, 2012. 1, 2, 6

[4] S. Hinterstoisser, V. Lepetit, S. Ilic, P. Fua, and N. Navab. Dominant orientation templates for real-time detection of texture-less objects. In *Proc. CVPR*, pages 2257–2264, 2010. 1, 2

[5] C. Steger. Occlusion, clutter, and illumination invariant object recognition. *Intl Archives of Photogrammetry and Remote Sensing*, 34, 2002. 1

[6] G. Borgefors. Hierarchical chamfer matching: a parametric edge matching algorithm. *PAMI*, 10(6):849–865, 1988. 1, 2

[7] A. Thayananthan, B. Stenger, P. H. S. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. In *Proc. CVPR*, volume 1, pages I–127, 2003. 1, 2

[8] W. J. Rucklidge. Efficiently locating objects using the hausdorff distance. In *Proc. IJCV*, volume 24, pages 251–270, 1997. 1, 2

[9] C. F. Olson and D. P. Huttenlocher. Automatic target recognition by matching oriented edge pixels. *IEEE Trans. Image Processing*, 6(1):103–113, 1997. 1, 2

[10] F. Tombari, A. Franchi, and L. Di Stefano. Bold features to detect texture-less objects. In *Proc. ICCV*, pages 1265–1272, 2013. 1, 2, 3, 5, 6, 7

[11] G. Kim, M. Hebert, and S.-K. Park. Preliminary development of a line feature-based object recognition system for textureless indoor objects. In *Proc. ICAR*, pages 255–268, 2007. 1, 2, 3, 5

[12] P. David and D. DeMenthon. Object recognition in high clutter images using line features. In *Proc. ICCV*, pages 1581–1588, 2005. 1, 2, 3, 5

[13] M. Awais and K. Mikolajczyk. Feature pairs connected by lines for object recognition. In *Proc. ICAR*, pages 3093–3096, 2010. 1, 2, 3, 5

[14] D. Damen, P. Bunnun, A. Calway, and W. Mayol-cuevas. Real-time learning and detection of 3d texture-less objects: a scalable approach. In *Proc. BMVC*, pages 23.1–23.12, 2012. 1, 2, 3, 5

[15] M. Muja and D. G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *Proc. VISAPP*, pages 331–340, 2009. 2, 6

[16] S. Procter and J. Illingworth. Foresight: fast object recognition using geometric hashing with edge-triple features. In *Proc. ICIP*, pages 889–892, 1997. 2

[17] R. C. Nelson and A. Selinger. A cubist approach to object recognition. In *Proc. ICCV*, pages 614–621, 1998. 2

[18] E. Hsiao and M. Hebert. Occlusion reasoning for object detection under arbitrary viewpoint. In *Proc. CVPR*, pages 3146–3153, 2012. 2, 6, 7

[19] M. Y. Liu, O. Tuzel, A. Veeraraghavan, and R. Chellappa. Fast directional chamfer matching. In *Proc. CVPR*, pages 1696–1703, 2010. 2

[20] M. Ulrich, C. Wiedemann, and C. Steger. Combining scale-space and similarity-based aspect graphs for fast 3d object recognition. *PAMI*, 34(10):1902–1914, 2012. 2

[21] B. Drost and S. Ilic. 3d object detection and localization using multimodal point pair features. In *Proc. 3DIMPVT*, pages 9–16, 2012. 2

[22] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of adjacent contour segments for object detection. *PAMI*, 87(3):284–303, 2007. 2

[23] V. Ferrari, F. Jurie, and C. Schmid. From images to shape models for object detection. In *Proc. CVPR*, pages 284–303, 2009. 2

[24] F. Jurie and C. Schmid. Scale-invariant shape features for recognition of object categories. In *Proc. CVPR*, volume 2, pages II90–II96, 2004. 2

[25] O. Carmichael and M. Hebert. Shape-based recognition of wiry objects. *PAMI*, 26(12):1537–1552, 2004. 2

[26] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 24(4):509–522, 2002. 2

[27] R. G. von Gioi, J. Jakubowicz, J. M. Morel, and G. Randall. Lsd: a fast line segment detector with a false detection control. *PAMI*, 32(4):722–732, 2010. 3

[28] A. Desolneux, L. Moisan, and J. M. Morel. *From gestalt theory to image analysis: a probabilistic approach*. ISBN: 0387726357. Springer, 2008. 3

[29] R. G. von Gioi, J. Jakubowicz, J. M. Morel, , and G. Randall. Lsd: a line segment detector. In *Proc. IPOL*, volume 2, pages 35–55, 2012. 3

[30] J. ORourke. Finding minimal enclosing boxes. *International Journal of Computer and Information Sciences*, 14(3):183–199, 1985. 3

[31] F. Perazzi, P. Krhenbhl, Y. Pritch, and A. Hornung. Saliency filters: contrast based filtering for salient region detection. In *Proc. CVPR*, pages 733–740, 2012. 3

[32] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: an efficient alternative to sift or surf. In *Proc. ICCV*, pages 2564–2571, 2011. 6, 7

[33] V. Chandrasekhar, D. M. Chen, S. S. Tsai, N.-M. Cheung, G. Takacs H. Chen, Y. A. Reznik, R. Vedantham, R. Grzeszczuk, J. Bach, and B. Girod. The stanford mobile visual search data set. In *Proc. MMSys*, pages 117–122, 2011. 6