This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the version available on IEEE Xplore.

BIND: Binary Integrated Net Descriptors for 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 BIND (Binary Integrated Net Descriptor), a texture-less object detector that encodes multi-layered binary-represented nets for high precision edge-based description. Our proposed concept aligns layers of object-sized patches (nets) onto highly fragmented occlusion resistant line-segment midpoints (linelets) to encode regional information into efficient binary strings. These lightweight nets encourage discriminative object description through their high-spatial resolution, enabling highly precise encoding of the object's edges and internal texture-less information. BIND achieved various invariant properties such as rotation, scale and edge-polarity through its unique binary logical-operated encoding and matching techniques, while performing remarkably well in occlusion and clutter. Apart from yielding efficient computational performance, BIND also attained remarkable recognition rates surpassing recent state-of-the-art texture-less object detectors such as BORDER, BOLD and LINE2D.

1. Introduction

Texture-less objects are a familiar sight in the real world and yet, widely established recognition algorithms such as SIFT (*Scale Invariant Feature Transform*) [1], and SURF (*Speeded Up Robust Features*) [2] are largely ineffective in such instances. This is due to their heavy reliance on highly-textured, feature-rich informative local regions, which are relatively scarce in homogeneous occurrences. Therefore, this challenging problem has led to various recent works such as [3–5], where object-sized regional information is exploited to gather discriminative content. Although producing decent results, extensive use of such sizable spatial architecture often leads to expensive computational run-times as well as memory load. As modern contemporary technologies such as real-time detection systems and mobile applications have limited computational resources, there is a growing consensus for today's algorithms to be robust, fast and compact.

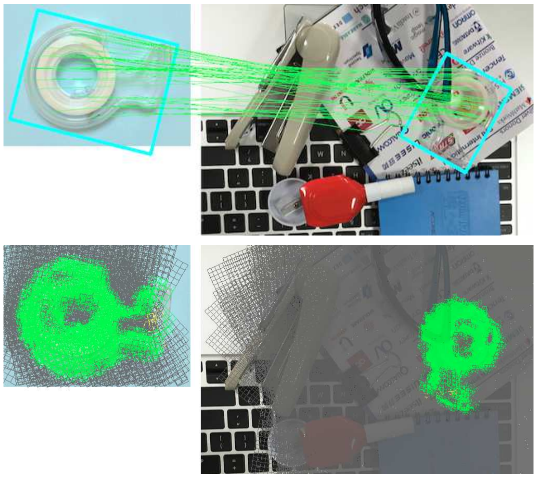


Fig. 1: BIND's texture-less object recognition in high clutter and semi-occlusion. (Top) keypoint matches, (bottom) binary net matches where green boxes represent angular block matches, yellow boxes indicate internal homogeneous block matches, and gray boxes refers to the background/no-match blocks.

A proven way to produce quick and memory efficient detectors without hardware acceleration is by means of binary descriptors. Works such as BRIEF (*Binary Robust Independent Elementary Features*) [6] and ORB (*Orient FAST and Rotated BRIEF*) [7] utilize local patches for simple binary intensity tests between pixels to form their descriptors. This led to very efficient vector sizes and matching speeds, as these strings can be quickly compared using the Hamming distance. However, these binary descriptors share the same predicament as the aforementioned algorithms, as they similarly require rich local information for their intensity tests.

Therefore, in pursuit of a texture-less detector that caters to the needs of modern technologies, we propose BIND (*Binary Integrated Net Descriptor*), a detector that describes in homogeneous conditions over large object-sized regions using uniquely designed multi-layered binary nets. BIND adopts a keypoint-descriptor structure by first forming interest points with *Linelets* [3], a recently introduced highly occlusion-resistant line-segment detector, followed by overlaying of our proprietary binary net-layers, oriented according to each linelet midpoint for encoding various regional characteristics of the object. Nets are essentially large object-encompassing square patches that are divided equally into high-resolution bit-representing blocks. Upon keypoint alignment, they transform into n net-layers to represent n-bit data with each internal block embedding information such as rotation-invariant angle primitives, along with the structural positioning of the object's edges and internal homogeneous space. BIND also addresses the problem of background contrast, whereby line-gradient directions vary in polarity due to diverse backdrops in the scene, which affects line/orientation edge-based descriptors such as [3, 4, 8]. Finally, matching is conducted through a series of binary arithmetic instructions, while incorporating various techniques to include enhancement properties such as scale invariance and occlusion resistance. In all, BIND's lightweight description technique not only yields efficient computational performance, but also delivers exceptional recognition competence in clutter and occlusion through its ability to isolate and sample regional object information at high spatial resolutions. Fig. 1 exhibits BIND's texture-less object recognition capability in a challenging scene.

The rest of the paper is presented as follows. Section 2 reviews related texture-less based works. Sections 3 and 4 describe the elements of BIND and its encoding schemes respectively. Section 5 summarizes the overall object recognition pipeline. Section 6 exhibits the comparative experimental results. Finally, section 7 concludes the paper.

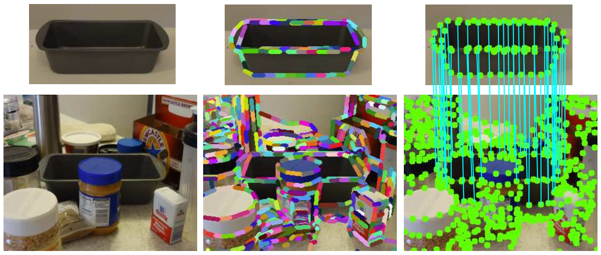
2. Related Work

Although countless object detectors have been proposed, very few can claim to robustly detect texture-less objects. Thus, this section reviews various techniques that had significant contributions to the texture-less genre.

Template-Based Detectors One pioneering approach that has the capacity to detect texture-less objects is Chamfer Matching [9, 10]. It is essentially an edge-based template matcher where detected contours between the model and scene are compared through a distance transform based dissimilarity measure. However, it is plagued with issues such as noise sensitivity and other occlusion factors. This raised several chamfer related enhancements such as shape-based [11], gradient-directional based [12, 13], and the Hausdorff distance based [14, 15] approaches with varying results. More recently, a gradient-based template approach by Hinterstoisser et al. [16] was introduced with reasonable success. Their technique coined LINE-2D, generates templates by quantizing gradient orientations into fixed directions, while adopting several optimization schemes for quick windowing similarity measures between input images. This approach gained modest popularity giving rise to various supplementary works such as [5, 17, 18] where features such as surface normals (LINE-MOD), color information, and occlusion reasoning were respectively added to aid in its development. Besides the aforementioned techniques, notable works such as a color/shape model [19] and a 3D-CAD based [20] template approaches also reported decent detection results. In spite of this, one major flaw that persists in template-based algorithms is scalability, where massive amounts of training data are often needed to compensate for the lack of visual properties such as rotation, viewpoint and scale changes.

Shape-Based Detectors Another technique proposed by Ferrari et al. [21, 22], groups local associated edge-chains called k-Adjacent Segments (kAS) to learn the object's shape model by consolidating its distances, orientations and lengths. This learned model then detects the scene object through an initial Hough voting localization followed by a shape matching algorithm within the voted area. This approach, or shape detectors in general, tends to be very sensitive to occlusion and minor distortions from interrupted or missing edges. Other related shape-based works include [23–26], where objects are trained into shape-based descriptors to enhance computational speeds and include properties such as scale-invariance.

Keypoint-Descriptor Detectors Among all of the approaches, keypoint-based techniques seemed to outperform the others due to its capacity to incorporate multiple invariant properties. In terms of recognition performance, BORDER (*Bounding Oriented-Rectangle Descriptor for Enclosed Regions*) [3] is the latest algorithm to achieve state-of-the-art stature. It garnered impressive detection results in heavily cluttered-occluded scenes through highly repeatable occlusion-resistant line-segments termed *Linelets*, which couples with a region encompassing oriented-rectangle revolution scheme for description. However, rectangle rotations can be relatively computationally expensive especially in cluttered scenes where large number of keypoints needs to be processed. Furthermore, its fairly-low block resolution [4×4] could affect descriptor precision in high-detailed object instances. Preceding BORDER is a line descriptor coined BOLD (*Bunch of Line Descriptor*) [4] where each line-segment [27] midpoint amasses neighboring segments for regional description. The gathered lines form angle primitive pairs to populate a two-dimensional descriptor. Although producing decent results, one caveat admittedly reported in its literature is the susceptibility to nearby clutter during line aggregations. Moreover, as reported in [3], line-segments in its original form, performs poorly due to midpoint deviations in occluding circumstances. Another significant work in this category includes an edgelet constellation technique by [8] whereby short segmented edges is accumulated using an angle-tracing path reflection method. However, this method is very sensitive to minor occlusion/illumination/noise as these cause alterations to the angle traces. Other algorithm of relevance to keypoint-based detectors include line-based works such as [28–30] where lines are associated and processed in various indifferent approaches.



(a) (b) (c)

Fig. 2: Maximal fragmented linelet detection for a texture-less object in an occluded setting. (a) Original model (top) and scene (Bottom) images. (b) Linelet detection with maximal fragmentation. (c) The highly repeatable, precise and occlusion resistant maximal linelet midpoint correspondences matched by manually placing the model image onto the scene image.

3. The BIND Elements

BIND commences its recognition scheme by adopting an occlusion resistant line-segment detector termed *Linelets*. These segments then materialize into keypoints and coalesce with our proprietary layered binary nets for description of a large object-sized region. Therefore, this section begins the BIND's methodology by detailing a couple of its basis elements, which lay the foundation for binary encoding and net-matching techniques.

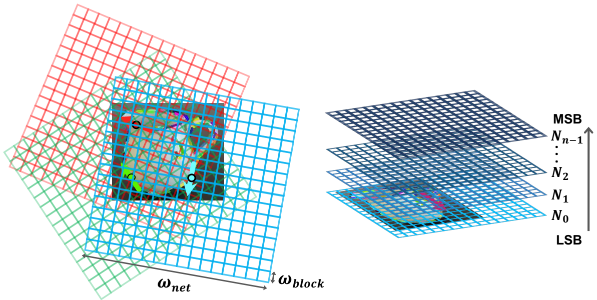
3.1. Maximal Linelet Fragmentation

Line-based representations have shown to be an effective approach for interest-point registration in the texture-less genre [3,4,8,28–30] as it provides a stable, repeatable and rotation-invariant platform for further descriptive purposes. Amongst these line-based techniques, *Linelets* [3], an extension of the *Line-Segments* [27, 31], has shown to be the most stable especially in occluding circumstances. It fragments overly elongated line-segments using a model-scene proportion concept by modulating their width according to the extent of clutter in the scene by,

$$\omega_\ell = \min[\max(\omega, R_{min}), \mathbb{L}_{max}], \qquad (1)$$

where ω_ℓ is the fragmentation width of line-segments that have grown beyond $2\omega_\ell$, while ω represents the width threshold derived from the detected line-segment ratios between the model-scene images [3]. $\mathcal{R}_{min}$ is a readily obtainable parameter that was hypothesized in LSD [31, 32] to automatically determine the minimum region size required to materialize any given cluster of closely oriented pixels as a line-segment. Lastly, $\mathbb{L}_{max}$ denotes the model object's longest line-segment, which reverts linelets back to line-segments when the model-scene proportion is low, i.e. $\omega > \mathcal{R}_{min}$ and $\omega \geq \mathbb{L}_{max}$. In all, linelets immunizes against segment midpoint shifts due to occlusion, whilst provisioning a highly repeatable basis for description.

Maximal Fragmentation As BIND emphasizes on a high-precision regional description concept, we propose to fragment linelets at its maximum frequency to accommodate the resolution of our descriptors. This can be simply accomplished by applying $\omega_\ell = R_{min}$ whenever a detected line-segment is $\geq 2\omega_\ell$. Although this adds computational load due to redundant fragmentations, our experiments (Fig. 8e) showed that even with the increased number of keypoints, BIND was still able to achieve competitive recognition speeds against other keypoint-based descriptors. Fig. 2 demonstrates the maximal-fragmented linelets' repeatability and precision for a texture-less object in an occluded and cluttered scene.



(a) (b)

Fig. 3: (a) Examples of (16 × 16) binary nets and their linelet alignments. (b) BIND's layering concept for each aligned net.

3.2. The Binary Net

To maximize distinctiveness, it has become customary for modern texture-less detectors to regionalize its descriptor scheme into the object-sized space [3, 4, 8, 16]. BIND however, goes a step further by not only encapsulating regions with large object-sized squared boxes, but also encouraging precise object description by heavily segmenting the box's internals to form its binary blocks. For each linelet keypoint, we "cast" this net by aligning its center onto the linelet midpoint and rotating it to the linelet's pointed direction. Regional information "captured" by the net is subsequently described by each internal block. Additionally, as binary representations only allow two possible states, we stack n additional net-layers to form up to 2^n states for encoding diversity. Fig. 3 demonstrates the binary net alignment for region encapsulation and BIND's net-layering concept, while the rest of this sub-section elaborates on its physical properties.

Block Size The squared divisions within the net are individual bit spaces that combine sequentially into a long binary string. It encodes the encapsulated homogeneous space, edge information, along with their chronological positions. To obtain the ideal balance between the net's encoding precision and overall capacity, we have designed the blocks to encapsulate all of the net-contained linelets at least once. This is achieved by assigning the block's width as $w_{block} = R_{Smin}$, where R_{Smin} is the minimum linelet length across all input models and their scales.

Net Size Rather than a standard-sized net, BIND designates its net dimensions to automatically conform to the input model object for optimal regional descriptiveness. This is achieved by first applying the minimum enclosing box algorithm [33] to the dataset-provided object mask, or the automatic-threshold salient mask [34] (Figs. 4a - 4c), and subsequently defining the initial net's width as $w'_{net} = 2l_{obj}$, where l_{obj} is the longer side of the current object enclosing box. In all, this design ensures ample coverage even when the net is situated at a far corner of the object.

Next, as the binary nets will be represented as bit strings, it is vital that the total blocks satisfy byte-formatting (multiples of 8) to accommodate computer storage and arithmetic techniques. Therefore, the total blocks per row/column and the finalized net width is established as,

$$n_{blocks} = 8 \cdot ceil\left(\frac{w'_{net}}{8 \cdot w_{block}}\right), \tag{2}$$

$$w_{net} = n_{blocks} \cdot w_{block}, \tag{3}$$

where the division between w'_{net} and w_{block} is rounded-up to the nearest multiple of 8 to find the total blocks per row/column n_{blocks}, and w_{net} is the final net width derived from n_{blocks} and w_{block}. In short, each binary net-layer's dimensions and total bit-count can be defined as $[n_{blocks} \times n_{blocks}]$. Note that although block width is fixed, net sizes and dimensions varies for each input model and scale.

4. Net Descriptor Encoding and Matching

This section first introduces the two main features that BIND describes within its nets, the object's internal homogeneous space, and its edges. Following that, binary net-layers are encoded into bit sequences through a carefully designed truth table, and finally matched using our unique logical operated techniques to incorporate resistive attributes such as scale, edge-polarity and occlusion.

4.1. The Internal Model Object Homogeneous Mask

One obvious feature that texture-less objects have in abundance is its internal homogeneous region. However, these blank spaces are often neglected by modern texture-less detectors [3, 4, 8, 16]. This lack of spatial differentiation between background and the internal object homogeneity could lead to many false positives due to distractors interacting with blank spaces in the scene. Thus, in BIND, we advocate the use of internal homogeneity as a key feature in our net description. This is done by generating a homogeneous mask $\mathbb{M}_H$ for accurate indication of the space within the model object. As demonstrated in Fig. 4, $\mathbb{M}_H$ is created by first applying a simple segmentation procedure (Fig. 4d) such as k-means color clustering [35], followed by iterating along the outer lines of the enclosing box [33] to gather background color labels (Fig. 4e), and finally assigning a '1' for any uncollected labels within the box to signify the object's internal anatomy (Fig. 4f). Note that this procedure was largely effective in our experiments due to good contrast between the model object and background for training, which is already a prerequisite for robust model description.

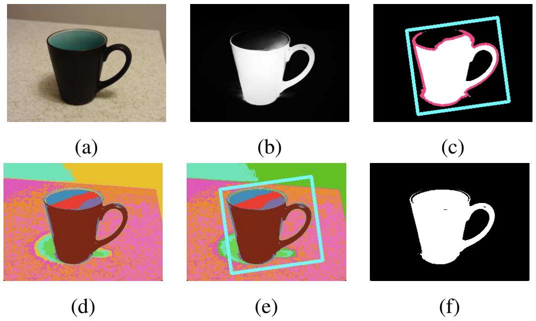


Fig. 4: The minimum enclosing box and homogeneous mask creation process. (a) The model image. (b) The model's saliency map. (c) Automatic thresholding of the salient map, and the minimum enclosing box (cyan) encapsulating detected contours (pink). (d) The k-means color cluster map. (e) The minimum enclosing box outer lines iterated to obtain the background clustered colors. (f) The final internal homogeneous mask derived by non-background colors within the enclosed box.

4.2. The Angle Primitive

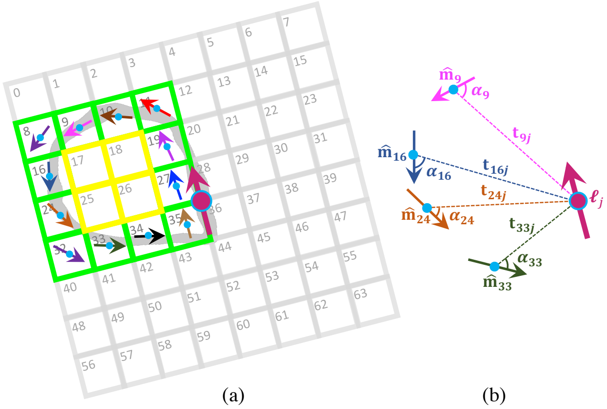
Arguably, the most consistent information that texture-less objects resonate is its noise and illumination impervious edge orientations. Similar to state-of-the-art works such as [3,4,8], BIND employs a pairwise line-based geometric technique in its descriptor-core to encrypt edge orientations into robust rotation invariant geometric primitives. This is realized by first detecting blocks that contain oriented pixels from the linelet creation clusters in section 3.1, followed by transforming these occupied block centers into unit vectors that point to the direction of its internally most influential linelet orientation, and finally pairing them with the origin linelet vector. This aggregation results in a vector-junction at the block center with various angles to choose from. State-of-the-art primitives used in both BOLD [4] and BORDER [3] computes an angle based on the unit vector's gradient-direction for added directional distinctiveness. However, one major pitfall of embedding the vector orientation directions into primitives is the susceptibility to polarity shifts, whereby contrasting backgrounds at different regions of the object causes gradients to point in the opposite direction. As this actuates corruption and degrades descriptiveness, BIND has opted for a line-direction invariant primitive to always take the smaller (acute) angle of the conjoint line pairs using,

$$\alpha = \arccos\left(\frac{|\hat{\mathbf{m}}_i \cdot \mathbf{t}_{ij}|}{\|\mathbf{t}_{ij}\|}\right), \tag{4}$$

where $\cdot$ represents the dot product, $\hat{\mathbf{m}}_i$ indicates the unit vector of the block's midpoint that is direction-influenced by its most infiltrated linelet, $\mathbf{t}_{ij}$ refers to the conjoining line between the origin net-linelet's midpoint ℓ_j, and $0^\circ \leq \alpha \leq 90^\circ$ refers to the smaller angle between $\hat{\mathbf{m}}_i$ and $\mathbf{t}_{ij}$ that represents the final line-direction invariant angle primitive, which will eventually be binary encoded to create BIND. Fig. 5 illustrates the transformations of the midpoint blocks into the line-directional invariant angle primitives.

4.3. Encoding the Binary Nets

As mentioned in section 3.2, binary nets are stacked into n net-layers to form 2^n binary states upon association



(a) (b)

Fig. 5: (a) Example of an 8 × 8 binary net encapsulating an object region. Each occupied block's midpoint vector points to the most influential linelet direction within it. (b) 4 randomly chosen polarity-invariant angle primitive examples between the origin linelet midpoint to the blocks 9, 16, 24 and 33 respectively.

with a linelet. Therefore, it is paramount to determine the total states needed for optimal descriptor distinctiveness and memory management. In BIND, we have designed and experimentally verified[1] that using 3 net-layers (N_2, N_1, N_0) to encode 8 states of information is the most efficient, as more layers not only adds memory load, but also increases the effects of quantization. The following paragraphs summarize the bit-combination for each state, and Table 1 details the BIND's 3-layer net design.

The Homogeneous State (Model Only) Whenever a block encapsulates an empty object space as labeled by the model's internal homogeneous mask $\mathbb{M}_H$ from section 4.1, all 3 net-layers at the particular block location are assigned as '1's ('111'). This is only encoded in model net-layers to indicate the object's internal homogeneous space, which is used for empty space comparisons with the scene net-layers for occlusion hypothesis during matching.

The Blank State For non-object areas, any external empty block location as indicated by the model's internal homogeneous mask is assigned as all '0's ('000'). This also applies to all scene's empty blocks as any empty space will simply be treated as a blank state.

The Angle Primitive States For a block that is occupied by oriented pixels, a unit vector would eventually culminate at its center to form the angle primitive α as described in section 4.2. To transform α into a binary state, it is quantized into 6 evenly distributed $\frac{\pi}{12}$ angle ranges within its angle limit of $[0, \frac{\pi}{2}]$, with each angle range assigned to one of the 6 binary states ('001' to '110') accordingly.

Descriptor Storage Structure Due to BIND's large net design, blank states would always overwhelm the other states, about 70-80% more on average. Therefore, to reduce the redundant space used for encoding blank states, BIND adopts a byte-indexing storage structure for each net-layer whereby only informative bytes (8 consecutive blocks that contain at least one non '000' state) is stored along side its byte index in a pairwise structure. Overall, this structure provides significant reduction in descriptor storage (about 50%) and match speeds due to lesser blank state iterations with no impact on BIND's recognition performance.

Table 1: BIND's 3-layer binary net bit-combinations. The net-layers' block-index locations $N_2(i)$, $N_1(i)$, $N_0(i)$ are assigned binary sequences based on its blocks' occupancy condition(s). An empty block renders a null $\alpha = \varnothing$, while an angle-occupied block is quantized to an angle range. The labels T and Q indicates whether a bit sequence is assignable to a Train or Query block respectively. Note that the final state '111', only applies to train blocks that has $\alpha = \varnothing$ and its center $C_b(i)$ indicating a $\mathbb{M}_H(C_b(i)) = 1$ in the model's homogeneous mask.

Image	Condition(s)	$N_2(i)$	$N_1(i)$	$N_0(i)$
T Q	$[\alpha(i) = \varnothing] \wedge [\mathbb{M}_H(C_b(i)) = 0]$ $\alpha(i) = \varnothing$	0	0	0
T, Q	$0° \leq \alpha(i) < 15°$	0	0	1
T, Q	$15° \leq \alpha(i) < 30°$	0	1	0
T, Q	$30° \leq \alpha(i) < 45°$	0	1	1
T, Q	$45° \leq \alpha(i) < 60°$	1	0	0
T, Q	$60° \leq \alpha(i) < 75°$	1	0	1
T, Q	$75° \leq \alpha(i) \leq 90°$	1	1	0
T	$[\alpha(i) = \varnothing] \wedge [\mathbb{M}_H(C_b(i)) = 1]$	1	1	1

4.4. BIND Matching

To compare bit-sequences between the model and scene net-layers, we apply various bit-wise logical operations to identically numbered layers and byte indexes before culminating into a single bit-string for total bit-count scoring. However, to alleviate occlusion resistance, a prior occlusion evaluation using the model's homogeneous state ('111') and the scene's angle primitive states is incorporated to prevent false positives from heavily textured scenes. This occlusion assessment score is determined by,

$$\mathbb{O}_{\mathcal{H}} = (\mathbb{T}_2 \wedge \mathbb{T}_1 \wedge \mathbb{T}_0) \ \wedge \ (\mathbb{Q}_2 \vee \mathbb{Q}_1 \vee \mathbb{Q}_0), \tag{5}$$

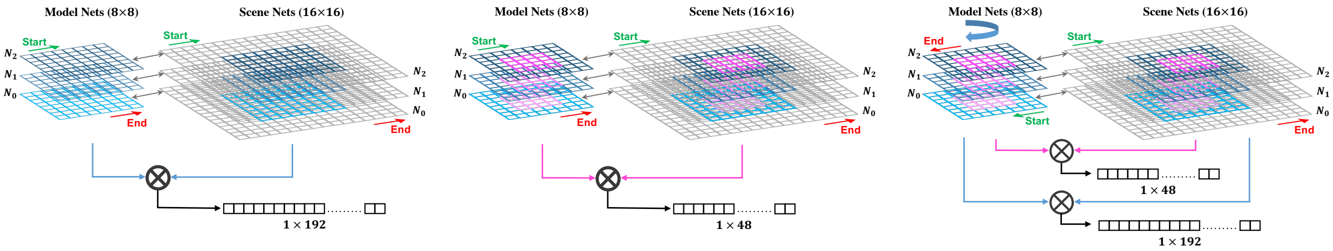
$$\mathcal{S}_{\mathbb{O}_{\mathcal{H}}} = \sum_{i=0}^{k-1} \mathbb{O}_{\mathcal{H}}(i), \tag{6}$$

where $\wedge$ and $\vee$ represent the array-wide bitwise AND and OR respectively, $\mathbb{T}_2$, $\mathbb{T}_1$, $\mathbb{T}_0$ and $\mathbb{Q}_2$, $\mathbb{Q}_1$, $\mathbb{Q}_0$ refer to the layers of a particular train and query net scheduled for comparison, the single layer output $\mathbb{O}_{\mathcal{H}}$ with the total block index size k indicating a '1' for each train internal homogeneous location that contains a query angle primitive block instead, and the total occlusion score $\mathcal{S}_{\mathbb{O}_{\mathcal{H}}}$ revealing the total occluded blocks in this particular net comparison. A sufficiently low occlusion score then activates the angle primitive scoring process by,

$$\mathbb{O}_{\mathcal{P}} = (\mathbb{T}_2 \odot \mathbb{Q}_2) \wedge (\mathbb{T}_1 \odot \mathbb{Q}_1) \wedge (\mathbb{T}_0 \odot \mathbb{Q}_0), \tag{7}$$

$$\mathbb{M}_{\mathcal{P}} = (\mathbb{T}_2 \vee \mathbb{T}_1 \vee \mathbb{T}_0) \wedge (\mathbb{Q}_2 \vee \mathbb{Q}_1 \vee \mathbb{Q}_0), \tag{8}$$

[1] Figure available in BIND's database. See page 8's footnote



(a) Scale Invariant Matching

(b) Occlusion net-reduction matching

(c) Polarity invariance matching

Fig. 6: Example BIND's matching procedure for (a) scaling, where a fixed-sized scene 16 × 16 3-layer net is matched with a scaled-down 8 × 8 net to produce a single binary string, (b) occlusion resistant procedure, where the model net reduces its section subset to match with the similar-sized midsection of the large scene net, and (c) polarity inversion, where all net comparisons are matched in 2 directions.

$$\mathbb{S}_{\mathcal{P}} = \begin{cases} \dfrac{\sum_{i=0}^{k-1} \left[\mathbb{O}_{\mathcal{P}}(i) \wedge \mathbb{M}_{\mathcal{P}}(i)\right]}{\mathcal{P}_{model}}, & \mathcal{S}_{\mathbb{O}_{\mathcal{H}}} \leq \lambda h_{model}, \\ 0, & otherwise \end{cases} \quad (9)$$

where $\odot$ represents the XNOR or the "equivalence" bitwise operator, $\mathbb{O}_{\mathcal{P}}$ refers to the 2D single layer angle primitive matched output, $\mathbb{M}_{\mathcal{P}}$ is a mask to ensure that the output score only includes angle primitive blocks, $\mathcal{P}_{model}$ indicating the current input model net's oriented-blocks ($\alpha(i) \neq \varnothing$), used to normalized total primitive score to obtain $\mathbb{S}_{\mathcal{P}}$. The manual-adjusted parameter termed the occlusion factor $0 < \lambda \leq 1$, and the total homogeneous model net blocks h_{model}, essentially regulate matches to heavily textured areas, while allowing some margin for minor occlusion in homogeneity. Although BIND enables λ for flexible adjustments, it is best specified according to the expected occlusion of the object in scene. However, as a reference, we have found that $\lambda = 0.5$ maintains a good balance in all of our experimental databases.

Unlike conventional descriptors, BIND varies its descriptor dimensions for each input model-scene pair. This was defined in section 3.2, where block widths w_{block} stay constant, whereas net widths w_{net} conform to twice the model's longer side. This unique framework enables BIND to incorporate various invariant properties through a technique we call, *sectional-extraction* matching, where the scene keypoints are described using the largest-scaled model net, and models are trained and matched with smaller/equal net subsets of the large scene net.

Scale Invariance To handle scaling, the model is downsized from its largest to form a pyramid, with linelet detection repeated at each level to re-align keypoints due to line disappearances/shifts at varying scales. This downsizing trend also creates several smaller subsets of the largest net, which is matched through the *sectional-extraction* technique. Fig. 6a shows a downsized model net matched to a subset of the large scene net.

Occlusion Handling For matching instances with high occlusion such that the homogeneity or angle primitive scores are undesirable, i.e. $\mathcal{S}_{\mathbb{O}_{\mathcal{H}}} > \lambda h_{model}$ or $\mathbb{S}_{\mathcal{P}} < \lambda$, BIND proceeds to re-iterate the comparisons from Eqs. 5-9 with reduced sectional subsets to obtain better scores. We have designed net iterations with $\Omega = \{0.75, 0.5, 0.25\}$ of their current model net scale, but only with the condition that these smaller subsets contain sufficient oriented-blocks, i.e. $\mathcal{P}'_{\Omega model} > \Omega\mathcal{P}_{model}$, where $\mathcal{P}'_{\Omega model}$ is the reduced model net block's current total oriented-blocks that replaces $\mathcal{P}_{model}$ in Eq. 9 for each Ω iteration. Overall, this enables information rich object regions to be matched even in high occlusion. Examples of the reduced sectional matching for occlusion handling can be seen in Figs. 6b and 7.

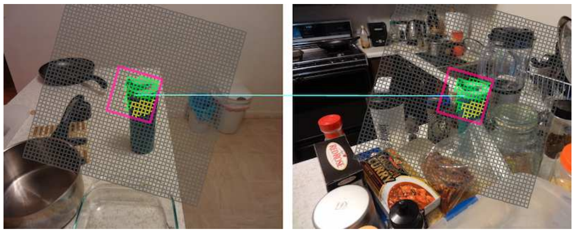
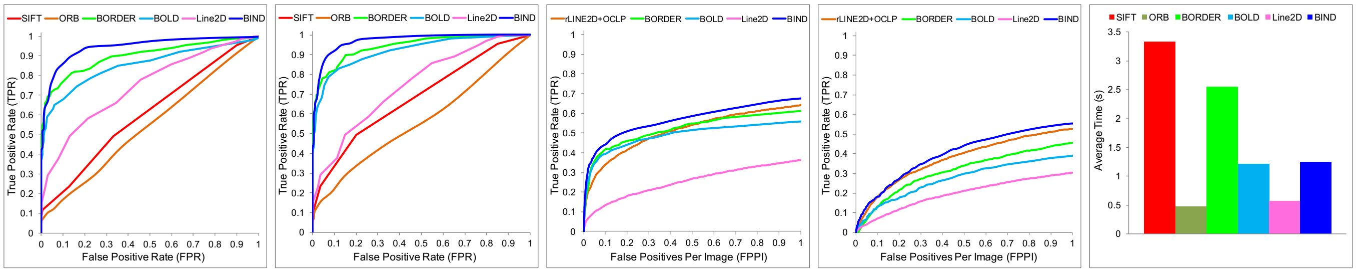


Fig. 7: An example of a net match with *sectional-extraction* where the occlusion is found to be too high. The pink box represents the (0.25) sectional extraction subset, and the green, yellow and gray boxes signifies the oriented, internal homogeneous, and background/no-match blocks respectively.

Polarity Invariance Mentioned in section 4.2, linelet directions are highly susceptible to polarity shifts due to background contrast especially in complicated scenes. Therefore, as the linelet-net alignment in section 3.2 was in a single direction, we counteract this by reversing the model's block sequence and re-applying Eqs. 5-9 for each net comparison to take the better score of the two directions. This 2-way matching procedure is practiced in all net comparisons, and likewise applies to all sectional reduced-net matchings. Fig. 6c shows the 2-way matching procedure for polarity invariance.

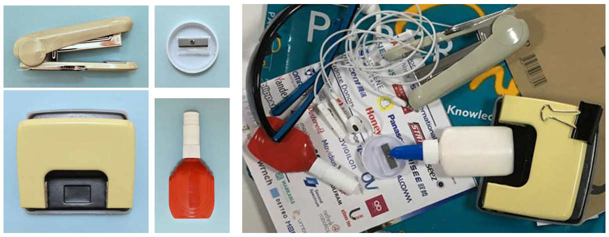
5. The BIND Object Recognition Pipeline

This section outlines the pipeline of BIND for the recognition of texture-less objects. The algorithm begins by detecting linelet midpoints (section 3.1), which are used as keypoints for its high precision and occlusion-resistant properties. This is applied onto a multi-scaled pyramid of model images, but only once for the original scene



(a) MOD (b) D-Textureless (c) CMU-KO8 Single (d) CMU-KO8 Multi (e) Average Runtimes

Fig. 8: Results of all experimental databases including the average time per image for the detectors.



(a) (b)

Fig. 9: *MOD*'s sample model images (a) and a scene image (b).

image. Following that, the enclosing boxes from each pyramid level form the model binary nets (section 3.2), while assigning the largest net size for scene description. Next, each level of the model pyramid generates its own individual internal homogeneous mask (section 4.1) for texture-less space encoding. Subsequently, nets are then triple-layered, direction-aligned to each keypoint, and have their blocks identified for internal homogeneity and angle primitives (section 4.2) before binary encoding using Table. 1 (section 4.3). Finally, everything comes together in the matching phase (section 4.4) where the model nets at each scale are compared with the large scene nets, and validated using geometric verification techniques [36, 37].

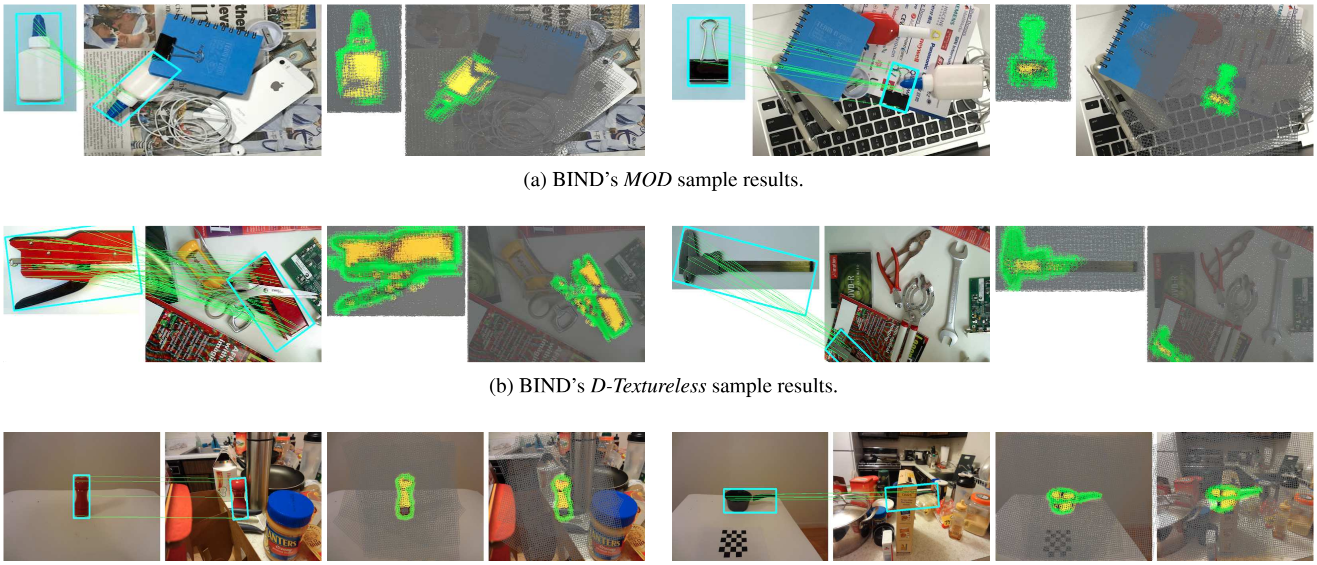
6. Experiments and Evaluation

To assess BIND's overall competence in texture-less object recognition, we have employed several algorithms with close relation to its qualities. For techniques that specialize in texture-less object detection, we engage state-of-the-art works such as the line-based approach of BORDER [3] and BOLD [4], together with the template matching-based scheme of LINE-2D [16]. Moreover, as BIND can also be categorized as a binary keypoint-based detector, we have included standard detectors such as SIFT [1], and its binary-based alternative ORB [7]. All mentioned algorithms including BIND, were implemented using an Intel dual-core i7 Haswell processor with 8GB of memory, and coded in C++ with their default libraries and recommended settings as provided in their respective works. A total of 3 texture-less object databases were evaluated, with two taken publicly and one self-contributed due to limited options in the texture-less object category.

The Messy Office Desk Dataset Coined *MOD* for short, this database assembled by our team simulates scenes of objects with high homogeneity placed around common workstations. It contains 9 models that randomly feature simultaneously within 100 scenes. Its aim is to appraise algorithms in a real environment on attributes like rotation, scale, translation, and distinctive properties such as clutter and occlusion. In addition, objects in various scenes of *MOD* are placed in random tonal backgrounds to challenge algorithms in complicated surroundings. For this experiment, we consolidate all the mentioned detectors' outcomes in an ROC plot as presented in Fig. 8a. Upon analysis, local descriptors like SIFT and ORB, produced below par performance, whereas the texture-less based solutions excelled with BIND championing the overall experiment. Although BORDER, BOLD and LINE-2D also had decent performances, a clear distinction between BIND can be observed especially in scenes where object edge-gradients are disordered by different regional backgrounds. This is mainly due to their high dependence on gradient-direction for edge description, while BIND's polarity invariant properties immunized itself to such conditions. Fig. 9 exhibits some of the *MOD* database models and a sample scene, while Fig. 10a presents some of BIND's recognition results in the *MOD* dataset.

The D-Textureless Dataset This database by the creators of BOLD [4], consists of 9 model and 55 scene tool-based images. Each scene image contains multiple models that examines algorithms on properties such as rotation, scale, translation, occlusion and clutter. We evaluated all the above mentioned algorithms and consolidated their ROC curves to yield the graph shown in Fig. 8b. As anticipated, texture-less detectors clearly outperforms the others, while BIND achieved the finest results, outperforming both BORDER and BOLD by a fair margin. Head-to-head analysis revealed that BIND tends to perform better than BORDER in circumstances that require high precision, and BOLD in situations with nearby clutter. This can be mainly attributed to BIND's highly-descriptive net design, enabling encoding precision to both the object's homogeneous space and its edges. Fig. 10b demonstrates the precision and occlusion/clutter resistance of BIND in this dataset with net-matching included.

The CMU-KO8 Dataset Also known as the *CMU Kitchen Occlusion Dataset* by Hsiao and



(a) BIND's *MOD* sample results.

(b) BIND's *D-Textureless* sample results.

(c) BIND's *CMU-KO8* sample results.

Fig. 10: Example recognition results of BIND in all three experimental datasets. For each case, keypoint matches along with the object detection bounding box results are presented on the left, and the net matches with green blocks indicating oriented blocks, yellow signifying internal homogeneous blocks, and gray implying background/no-match blocks are displayed on the right.

Hebert [18], this database was assembled to evaluate their occlusion-reasoning add-ons for LINE-2D to improve its performance in heavy clutter and occlusion. For that reason, this dataset mainly focused on placing texture-less objects in highly distracted scenes, with virtually no attention was placed on variances such as rotation and scale. In all, this database contains 8 kitchenware models alongside 2 scene branches of 800 single point-of-view images, and 800 multi-view images where each scene image contains only one instance of a model object. All mentioned algorithms including the best resulting occlusion model in [18] coined rLINE2D+OCLP (Occlusion Conditional Likelihood Penalty) were assessed up to the detection rate of 1.0 FPPI in the plots shown in Figs. 8c and 8d. Note that due to poor registration rates of SIFT, SURF and ORB, their results were not included in the plots. Overall, BIND attained the finest detection rates outperforming the state-of-the-art BORDER and even rLINE2D+OCLP in its own dataset. Closer inspection revealed that BIND's high net precision and occlusion descriptive measures were the definitive factors that facilitate its superior true positive rates in such challenging scenes. Fig. 10c exhibits some of BIND's keypoint and net recognition results from this database.

Timing Comparison As BIND is predominantly a binary-based detector, it has naturally fast matching speeds. However, it is somewhat hindered by its highly precise linelet/description process in the experiments, attaining similar recognition speeds as BOLD, and about 2.2 times faster than the state-of-the-art BORDER as presented in Fig. 8e. Even so, BIND can be easily customizable for speed-ups at the expense of precision to attain real-time detection.

Memory Comparison In our experiments, full-scale objects typically converge to a [48 × 48] dimensional net, granting a 30 times precision increase in informative block counts, and about 50-100 bytes (more at smaller pyramid levels) savings in terms of descriptor size (with BIND's storage structure), when compared to BORDER's [4 × 4] 128-dimension [3], and BOLD's 2D 12-bin histogram [4] floating vector descriptors respectively.

7. Conclusion

BIND, a multi-layered net-based binary descriptor for texture-less object recognition is proposed in this paper. It provides precise regional object description through a triple-layered net design to encode edges and internal homogeneous spaces into compact rotation-invariant binary strings, while inspiring attributes such as polarity, scale and occlusion resistances in its matching phase. In all, experiments from three databases[2] showcased BIND's overwhelming robustness in recognizing texture-less objects in a wide variety of situations. However, as BIND was strictly designed for texture-less objects through its 3-layer bit combination table, it is not as effective in low-homogeneity unlike its counterparts (e.g. BORDER, BOLD, etc.). Nevertheless, we see BIND's underlying binary net-layering concept as a strong basis for many other recognition tasks through the redesigning potential of its layers/tables to incorporate various indifferent attributes.

This research was partially supported by National Research Foundation, Prime Ministers Office, Singapore under its IDM Futures Funding Initiative and AcRF Tier 1 (RG28/15).

[2] Full experimental results and video demo available at: https://drive.google.com/open?id=0B-vEAVo5DHXFS0FqTlNkcGJvcEk

References

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

[3] J. Chan, J. A. Lee, and K. Qian. Border: An oriented rectangles approach to texture-less object recognition. In *Proc. CVPR*, 2016. 1, 2, 3, 4, 7, 8

[4] 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, 4, 7, 8

[5] 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] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. BRIEF: Binary Robust Independent Elementary Features. In *ECCV*, pages 778–792, 2010. 1

[7] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: an efficient alternative to sift or surf. In *Proc. ICCV*, pages 2564–2571, 2011. 1, 7

[8] 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. 2, 3, 4

[9] H. G Barrow, J. M Tenenbaum, R. C. Bolles, and H. C. Wolf. Parametric Correspondence and Chamfer Matching: Two New Techniques for Image Matching. *IJCAI*, pages 659–663, 1977. 2

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

[11] 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. 2

[12] C. Steger. Occlusion, clutter, and illumination invariant object recognition. *Intl Archives of Photogrammetry and Remote Sensing*, 34, 2002. 2

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

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

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

[16] 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. 2, 3, 4, 7

[17] X. Peng. Combine color and shape in real-time detection of texture-less objects. *Computer Vision and Image Understanding*, pages 31–48, 2015. 2

[18] E. Hsiao and M. Hebert. Occlusion reasoning for object detection under arbitrary viewpoint. In *Proc. CVPR*, pages 3146–3153, 2012. 2, 8

[19] J. I. Olszewska. Where is My Cup? - Fully Automatic Detection and Recognition of Textureless Objects in Real-World Images. *CAIP*, pages 501–512, 2015. 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] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of adjacent contour segments for object detection. *PAMI*, 87(3):284–303, 2007. 2

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

[23] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 24(4):509–522, 2002. 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] Y. Dou, M. Ye, P. Xu, L. Pei, and Z. Liu. Object Detection Based on Two Level Fast Matching. *International Journal of Multimedia and Ubiquitous Engineering*, 10(12):381–394, 2015. 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. 2, 3

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

[29] 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. 2, 3

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

[31] 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

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

[33] J. ORourke. Finding minimal enclosing boxes. *International Journal of Computer and Information Sciences*, 14(3):183–199, 1985. 4

[34] 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. 4

[35] D. E. Ilea and P. F. Whelan. Color image segmentation using a spatial k-means clustering algorithm. In *Proc. IMVIP*, 2006. 4

[36] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981. 7

[37] J. Chan, J. A. Lee, and K. Qian. F-sort: An alternative for faster geometric verification. In *Proc. ACCV*, 2016. 7