

# Vectorisation of Sketched Drawings Using Co-occurring Sample Circles

Alexandra Bonnici and Kenneth P. Camilleri

Department of Systems and Control Engineering, Faculty of Engineering,  
University of Malta

`alexandra.bonnici@um.edu.mt`, `kenneth.camilleri@um.edu.mt`

**Abstract.** This paper presents a drawing vectorisation algorithm which uses multiple concentric families of circles placed in a dense grid on the image space. We show that any off-centered junction within the family of circles can be located and hence show how these junction points may be linked to neighbouring junction points, thereby creating a vector representation of the drawing geometry. The proposed algorithm identified 98% of the junctions in the drawings on which it was evaluated, each within a localisation error of  $4.7 \pm 2.3$  pixels, resulting in straight line vectors which are well placed with respect to the drawn edges.

## 1 Introduction

Computer interpretation of sketches and drawings, in its simplest form, changes the hand-drawn, raster drawings into a vector format that can be used by computer-aided design tools [15, 21]. Drawings are more than the sum of individual lines and human observers use domain knowledge to identify particular arrangements of the line vectors and use this knowledge to interpret the drawings. These particular line arrangements may be used by drawing interpretation algorithms to extract structures from the drawing in a similar manner as a human expert would [3, 13]. In this work, we are particularly interested in the ability of inferring 3D shape from 2D drawings, where the structures of interest are the junctions that form the skeletal 3D structure of the sketched objects. Vectorisation techniques typically follow a two-tier approach, where the drawing is first binarised, following which, lines are sampled to obtain a vector representation of the drawing [15, 22, 5]. Such an approach incurs the additional costs of the binarisation step which may be problematic, particularly if the image background is of non-uniform intensity, such as what would be obtained in drawing images digitised by means of cameras on smart-phones or tablets.

In this paper, we propose an alternative vectorisation algorithm that can work directly on grey-level images. The image is sampled using circle samplers, labelling each sampled point as a line or junction point, where junction points can be further classified according to the specific junction geometry.

The rest of the paper is organised as follows: Section 2 presents a literature review on vectorisation algorithms; Section 3 presents our proposition of using

concentric circle samplers to detected off-centered junctions within the sampled region; Section 4 shows how the circle samplers can be used to vectorise the drawing; Section 5 presents the results obtained by the proposed algorithm while Section 6 concludes the paper.

## 2 Related Work

Converting line drawings into a vector format requires the localisation of lines from the image, discarding pixels that account for the stroke line widths retaining only those necessary to maintain the topology of the drawing, that is, the line's medial axis. This can be achieved through skeletonisation [9] which will however introduce distortions as the lines approach junction points [6]. To avoid these distortions, other methods can be employed to obtain the medial axis, for example, through the use of contours [22] and Delaunay triangles [16] among others. Since these methods assess whether each pixel is a candidate pixel on the medial axis, they incur large computational costs which may be reduced by adapting sparse pixel tracking approaches such as that described in [18, 17, 1] among others. Here, lines are extracted by sampling the image using rectangles [18], squares [17] and circles [1], starting with a strategically located seed pixel which is propagated along the lines of the drawing. Since lines have a rectilinear shape, rectangular samplers have the advantage of having a similar shape as the lines being sampled [12]. However, this will also require rotating the sampler so that it is aligned with the lines in the image [12]. This alignment requirement can be avoided by taking the grey-level profile of the boundary pixels [17, 1]. In such cases, circle samplers, being isotropic, are more suited samplers. Line location can also be performed by using the Hough transform, which accumulates the line parameters in an accumulator array [19]. Peaks in the accumulation array correspond to the image line parameters which must then be localised in the image space [19]. The grey-level image is typically pre-processed to obtain an edge-map, such that the peak localisation is dependent on the accuracy of the edge detection. Thus research is invested in more robust edge localisation [4].

Vectorization by sparse pixel tracking and the Hough transform techniques can extract lines in their entirety even in the presence of intersections with other lines. However, these intersections are valuable for the algorithmic, 3D interpretation of the drawing, since such intersections form the junctions which define the 3D geometry of the object. For example, line-labelling algorithms such as [14] among others, require the identification of the geometry of the junction points in order to provide meaningful line labels. To this extent, additional post-processing is often performed to detect line intersections [8, 10].

Sampling the grey-level profile of the image around a given line can potentially provide for the detection of junction points without the requirement of additional post-processing. In particular, the isomorphic nature of circle samplers allows the sampling of a line pixel neighbours in a locus of points equidistant from the line point, hence sampling any other line that can potentially intersect with the line being sampled. Sparse pixel tracking however requires sequential

tracking of the image, starting from some seed pixel centred on the line [18, 17], incurring additional computational overheads to locate a suitable seed and maintain the path propagation.

We hypothesise that the grey-level profile of the circle sampler can be used to deduce the orientations of lines passing through the circle as well as the location of a junction point within the circle such that the image can be sampled simultaneously, in its entirety by multiple circles strategically placed across the image.

### 3 Junction localisation from sampling circles

Consider a binary image containing a junction point  $P$  with position vector  $\mathbf{J}_P$ . The junction is formed when thin line segments with unknown orientations  $\theta$ ,  $\theta = [0, 2\pi)$  intersect at  $P$ . In the particular case, the junction has  $N$  line segments emerging from it, each with orientation  $\theta_n$ ,  $n = 1 \cdots N$ . A pixel position on any of the line segments may be expressed as:

$$\mathbf{x}_n(r) = \mathbf{J}_P + r\mathbf{k}_{\theta_n} \quad (1)$$

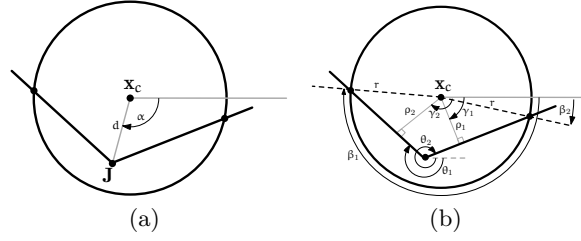
where  $\mathbf{x}_n(r)$  is the pixel coordinates of a point on the  $n^{\text{th}}$  line segment forming the junction,  $r$  is the displacement from the junction point and  $\mathbf{k}_{\theta_n} = [\cos \theta_n, \sin \theta_n]^T$ . If  $\theta \in [0, 2\pi)$ , then (1) describes a single point on the circumference of such a circle, specifically the point at which the line segment emerging from the junction point intersects with the circle circumference. Such a circle can be used to sample the image around the junction, in which case, the grey-level profile of the circle circumference can be ideally modelled by:

$$I_r(\beta(\theta)) = \begin{cases} 1 & \text{when } \beta(\theta) = \theta_n \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where  $\beta(\theta) = [0, 2\pi)$  is the angle on the circle circumference at which the line segment intersects with the circle circumference. This sampling circle can therefore be used to determine the line orientation of the line segments at the junction. Repeating the sampling by using concentric sampling circles increases the confidence in the estimated line orientations. Moreover, junctions consist of two or more line pairs which have an angular separation  $\Delta\theta$  such that we may define a co-occurrence matrix  $S(\theta, \Delta\theta)$  as:

$$S(\theta, \Delta\theta) = \frac{1}{M} \sum_m I_{r_m}(\beta(\theta)) I_{r_m}(\beta(\theta + \Delta\theta)) \quad (3)$$

where  $r_m$  is the radius of the  $m^{\text{th}}$  circle,  $m = 1 \cdots M$  and  $M$  is the total number of concentric sampling circles. This allows for further discrimination between line segments that are part of the centered junction and others that are accidental intersections with the sampling circles. In this particular case, with the circles centred on the junction,  $\beta(\theta) = \theta$  and the concentric circles centered on  $P$  will have identical grey-level profiles  $I_r(\beta(\theta))$ .



**Fig. 1.** A sampling circle containing an off-centred L-Junction showing (a) the junction parameters and (b) the line parameters and intersecting angles.

**Lemma 1.** *The line orientations at a junction may be estimated from  $[\hat{\theta}, \Delta\hat{\theta}] = \arg\{\max\{S(\theta, \Delta\theta)\}\}$*

*Proof.* Following from (2),  $I_{r_m}(\beta(\theta))I_{r_m}(\beta(\theta + \Delta\theta)) = 1$  if and only if the sampling circle intersects with two line segments with some orientation  $\theta$  and  $\theta + \Delta\theta$ . If these line segments form part of a centred junction, then for all  $M$  concentric sampling circles,  $\beta(\theta) = \theta_n$  and  $\beta(\theta + \Delta\theta) = \theta_n + \Delta\theta$ . If however the line segments do not form part of the centred junction and are just accidental intersections between the sampling circles, then  $\beta(\theta) \neq \theta_n$  and  $\beta(\theta + \Delta\theta) \neq \theta_n + \Delta\theta$ . Moreover, the angle of intersection between the sampling circles and the lines will be different for all sampling circles and hence,  $S(\theta, \Delta\theta) = \frac{1}{M}$ . Thus, as  $M \rightarrow \infty$ ,  $S(\theta, \Delta\theta) \rightarrow \mathbf{E}[I_r(\beta(\theta))I_r(\beta(\theta + \Delta\theta))]$  and hence,  $[\hat{\theta}, \Delta\hat{\theta}] = \arg\{\max\{S(\theta, \Delta\theta)\}\}$  is a consistent estimator of the line orientations of the line segments at a centred junction.  $\square$

### 3.1 Localisation of off-centered junction points

Although Lemma 1 shows that circle samplers can be used to determine the line orientations of all line segments at a junction, the circle samplers must be centered on the junction point, thus incurring computational overheads required to select suitable seed pixels to act as circle centres. In order to bypass this, it is desirable to locate the junction point when it occurs anywhere within the sampling circle, relaxing the requirement of having the set of concentric circle samplers centered on the junction point.

Consider the centre  $C$  of the circle sampler as a local point of reference. As shown in Fig. 1(a), the junction point  $P$  can be referred to  $C$  using the parameters  $\mathbf{d} = (d, \alpha)$ . Similarly, each junction line segment can be referred to the circle centre using the polar coordinates  $(\rho, \gamma)$  as shown in Fig. 1(b). The angle on the circumference at which the line segments intersect with the circle circumference is now a function of the junction position, the radius and the line orientations such that (2) may be re-written as:

$$I_r(\beta(r; \mathbf{d}, \theta)) = \begin{cases} 1 & \text{when } \beta(r; \mathbf{d}, \theta) = \gamma \pm \cos^{-1}\left(\frac{\rho}{r}\right) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where  $\rho = d \cos(\gamma - \alpha)$  and  $\gamma = \theta + \frac{\pi}{2}$ . Since line segments terminate at the junction,  $\beta(r; \mathbf{d}, \theta)$  is bound by  $\alpha$  and  $\theta$ , such that

$$\beta(r; \mathbf{d}, \theta) \mapsto \begin{cases} [\alpha, \theta] & \text{if } \alpha \leq \theta < \alpha + \pi \\ (\theta, \alpha] & \text{if } \alpha + \pi \leq \theta < \alpha \end{cases} \quad (5)$$

The co-occurrence matrix of (3) may therefore be extended to

$$S(\mathbf{d}, \theta, \Delta\theta) = \frac{1}{M} \sum_m I_{r_m}(\beta(r_m; \mathbf{d}, \theta)) I_{r_m}(\beta(r_m; \mathbf{d}, \theta + \Delta\theta)) \quad (6)$$

**Lemma 2.** *The junction position and line orientations may be estimated from  $[\hat{\mathbf{d}}, \hat{\theta}, \Delta\hat{\theta}] = \arg\{\max\{S(\mathbf{d}, \theta, \Delta\theta)\}\}$ .*

*Proof.* Given any sampling circle of radius  $r_m$ , there will be an infinite number of combinations of the parameters  $(\mathbf{d}, \theta, \Delta\theta)$  for which  $\beta(r; \mathbf{d}, \theta)$  corresponds to the intersecting angle between the line segment and the circle circumference. However, only if  $(\mathbf{d}, \theta, \Delta\theta)$  correspond to the junction position and line orientations will  $I(r_m, \beta(r_m; \mathbf{d}, \theta)) = 1$  for all  $m = 1, \dots, M$ . Otherwise,  $I_{r_m}(\beta(r_m; \mathbf{d}, \theta)) = 1$  only for specific sampling circles.

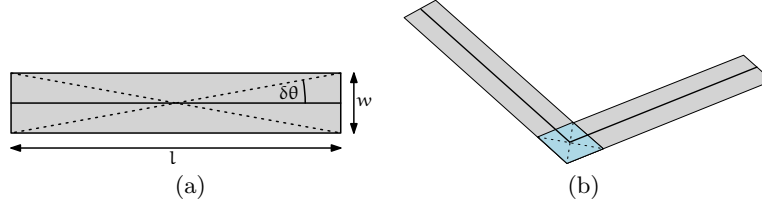
Hence, as  $M \rightarrow \infty$ ,  $S(\mathbf{d}, \theta, \Delta\theta) \rightarrow \mathbf{E}_r[I_r(\beta(r; \mathbf{d}, \theta)) I_r(\beta(r; \mathbf{d}, \theta + \Delta\theta))]$  and thus,  $\arg\{\max\{S(\mathbf{d}, \theta, \Delta\theta)\}\}$  is a consistent estimator of the junction position and line segment orientation.  $\square$

Note that in the particular case when the junction is centered on the circle samplers,  $\mathbf{d} = 0$  so that  $S(\mathbf{d}, \theta, \Delta\theta) = S(0, \theta, \Delta\theta) = S(\theta, \Delta\theta)$  as represented in (3).

### 3.2 Localisation of junctions with thick line segments

The discussion thus far assumes that the drawing consists of thin line segments. In practice, this is not the case since the resolution of the scanning device and the pen thickness introduce some finite line width  $w$  to the strokes. If we model the stroke segments as rectangular strips of length  $l$  and width  $w$ , as shown in Fig. 2(a), then it can be easily observed that the orientation resolution of the line segment is reduced to  $\theta \pm \delta\theta$  where  $\delta\theta = \tan^{-1}(\frac{w}{l})$ , corresponding to the lines passing through the diagonals of the rectangular segment. Thus, for a given junction position, there will be a range of values of  $\beta(r; \mathbf{d}, \theta)$  for which  $I_r(\beta(r; \mathbf{d}, \theta)) = 1$ , corresponding to an arc on the circumference of the sampling circle. Consequently, for some junction position vector  $\mathbf{d}$ ,  $S(\mathbf{d}, \theta, \Delta\theta)$  will contain regions of maximal values, centered on  $(\theta, \Delta\theta)$  and whose size is dependent on  $\delta\theta$ .

The thickness of the line segments will also reduce the localisation resolution of the junction point to a parallelogram region which can be described as the locus of points common to both line segments as shown in Fig. 2(b). The size and aspect ratio of this area are dependent on the line widths as well as the angular separation  $\Delta\theta$  between the lines at the junction. The diagonals of this



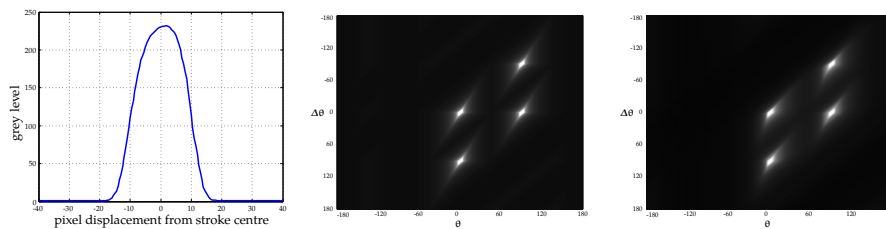
**Fig. 2.** (a) Thick lines can be modelled as rectangular segments of width  $w$  and length  $l$ . (b) The junction area formed by the two thick line segments. This can be described as the loci of points common to both line segments. Diagonals of the junction area intersect at the true junction point.

junction region will correspond to the true junction point and thus, for some value of  $(\theta, \Delta\theta)$ ,  $S(\mathbf{d}, \theta, \Delta\theta)$  will contain regions of maximal values centered on the junction point specified by  $\mathbf{d}$  that correspond to the junction location in the sampling circle. Thus  $\arg\{\max\{S(\mathbf{d}, \theta, \Delta\theta)\}\}$  will contain groups of  $(\hat{\mathbf{d}}, \hat{\theta}, \Delta\hat{\theta})$  values, each centred on the true values of  $(\mathbf{d}, \theta, \Delta\theta)$  such that the group centroid can be used as estimates for the line orientations and junction location estimates.

### 3.3 Junction and line estimates from grey-level images

Scanned images of drawings are often grey-level images rather than binary images and therefore the co-occurrence of the line stroke intersections with the circle sampler will be affected by the grey-level distribution of the line strokes. Apart from the anti-aliasing effect introduced by the digitising medium, resulting in image blur, the grey-level distribution at the sampling circle will also be affected by the nature of the pencil used to make the drawing strokes as well as the paper upon which the drawing is made. In general, a pencil can be characterised by the degree of hardness, the mixture of graphite, clay and wax particles in the writing core, the shape of the pencil tip and the pressure distribution of the tip [7, 20]. The shape of the tip can be modelled by polygons, with the area of the tip in contact with the paper will vary depending on the angle at which the pencil is held to the paper. In general, the midpoint of the polygonal shape can be considered as the main pressure distribution point, that is, the pencil will distribute most graphite onto the paper at the centre of the pencil tip and hence, the strokes produced by the typical pencil will be darker towards the centre of the stroke [20]. This can be observed in Figure 3(a) which shows the average stroke grey-level obtained from 30 strokes taken from 6 sample drawings.

Thus, line and junction parameters corresponding to line segments passing through the periphery of the stroke will not have co-occurrence values which are as strong as for those that correspond to line segments passing through the stroke centre. Consequently, the region of maximal values in  $S(\mathbf{d}, \theta, \Delta\theta)$  will be smaller than for crisp, binary strokes. These regions are still centered around the line and junction parameters in  $S(\mathbf{d}, \theta, \Delta\theta)$  such that the parameter estimation



**Fig. 3.** (a) The grey profile of a line stroke and the co-occurrence values for (b) a binary image with thick lines, (c) a pencil sketch of the image

is not affected by the grey-level images. This is advantageous to the vectorisation problem since the algorithm does not require prior binarisation.

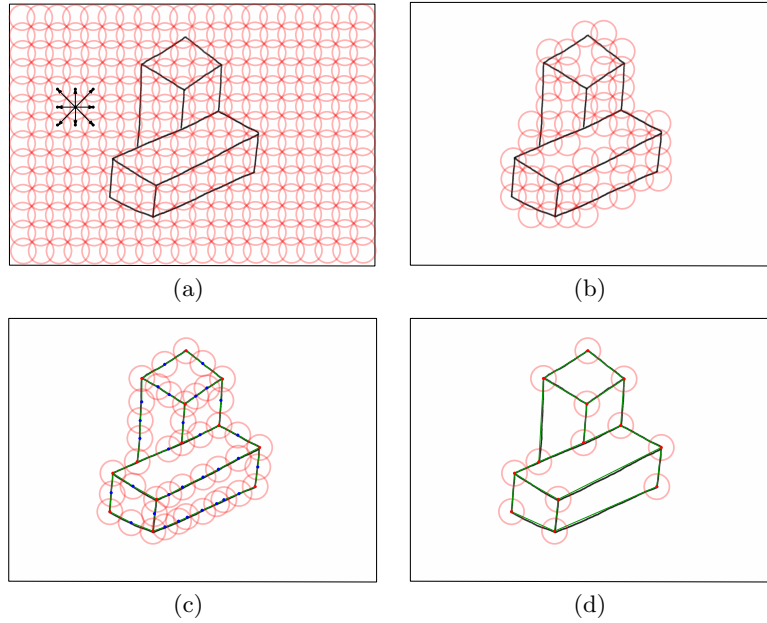
### 3.4 Sampling long, straight line segments

When sampling circles are placed at random locations within the image, it may be possible that although a line segment emerging from a junction passes through the circle area, the junction itself lies beyond the sampling circle. In such cases, junction point specified by  $\mathbf{d}$  that falls on the line segment,  $S(\mathbf{d}, \theta, \Delta\theta)$  will be a maximum when  $\theta$  corresponds to the line orientation and  $\Delta\theta = \pi$ . Thus, while  $\arg\{\max\{S(\mathbf{d}, \theta, \Delta\theta)\}\}$  will consist of a compact set for  $\theta, \delta\theta$ , there will be a wider range of possible values for  $\mathbf{d}$  parameter values, spanning the entire length of the line segment. The mean of these values will correspond to a point at the centre of the line segment and we define such a point as a *virtual* junction point.

## 4 Using concentric sampling circles for vectorisation

In order to vectorise the drawing, circle samplers should cover the whole image such that all junction points are sampled. These junction points must then be connected to obtain the topological structure of the drawing. Straight line segments are used between the detected junctions making up the topological structure to approximate the line drawing segments between the junctions. To ensure that the drawing is sampled completely, sampling circles are placed on an equidistant grid of centre points covering the entire image. The size of the grid is set such that it corresponds to the size of the largest inscribed square in the circle sampler, that is, the grid size is set at  $\sqrt{2}r_M$ , where  $r_M$  is the radius of the largest circle in the concentric set of sampling circles. This ensures that every junction point in the image will be sampled by at least one circle as shown in Fig. 4(a).

Within such a grid system, each sampling circle has eight neighbours to which it can be connected with a straight line vector. However, not all these circles necessarily contain line strokes and if a sampling circle samples only the image background, this can be discarded, removing the vector link that connects such

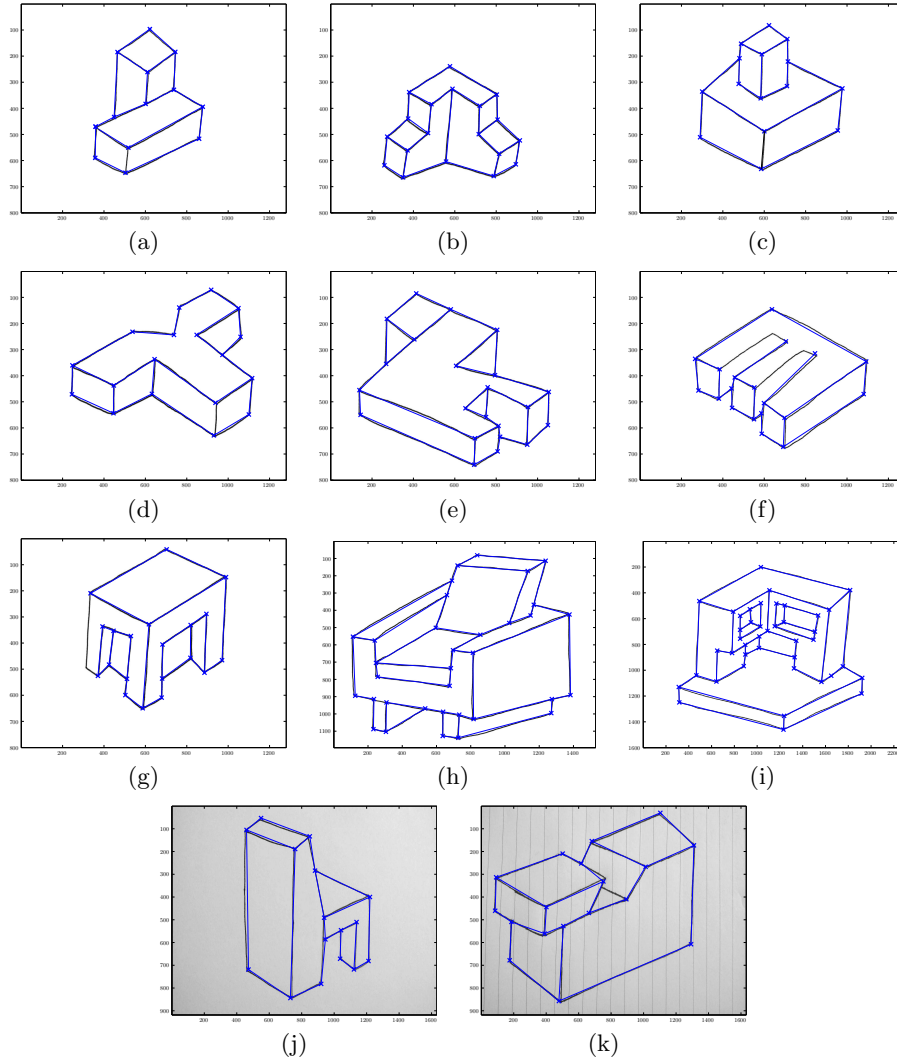


**Fig. 4.** Using sampling circles for vectorisation. (a) sampling circles are placed on an evenly spaced grid, allowing the sampling circles to overlap such that the entire image space is sampled. Each circle can be linked to eight neighbouring circles. (b) Sampling circles that fall on the image background can be removed, deleting redundant links accordingly. (c) Remaining circles can be aligned to the junction points located within the circles, merging samplers that correspond to the same junction point. (d) Virtual junction points may be removed, retaining only the true junction points and hence obtaining a vector representation of the drawing.

a circle to its neighbours as shown in Fig. 4(b). The remaining circles are centred on the junction point, refining the resolution of the line vectors. Moreover, since there is a degree of overlap between the sampling circles, circles that sample the same junction point are merged into a single circle which inherits the vector links of the parent samplers as shown in Fig. 4(c).

In this way, the topology of the drawing may be refined until all circles are centred on the junction points or the virtual junction points on the line medial axis. All circles corresponding to virtual junction points can then be removed, retaining the topological structure of the drawing by creating direct links between sampling circles that were initially linked through the virtual junction point. Thus, the resulting topology contains only junction points and vector links which correspond to the edges linking these junction points as shown in Fig. 4(d).





**Fig. 5.** The vectors obtained from the vectorisation algorithm superimposed on the drawings.

## 5 Results

The algorithm was evaluated on sample drawings such as those shown in Fig 5. These drawings vary in complexity, with drawings (a) - (f) showing simple geometric forms which provide examples of the junctions typically found in trihedral drawings while drawings (g) - (h) consist of more complex trihedral shapes. Drawings (j) and (k) were digitised using an in-built tablet camera, with drawing (j) drawn on plain white paper, while drawing (k) drawn on feint-ruled paper.

In this evaluation, the algorithm was implemented using a maximum sampling circle size of 100 pixels and 70 concentric sampling circles at each sampling site. Moreover, the maximum value obtained from  $S(\mathbf{d}, \theta, \Delta\theta)$  was required to exceed a maximum value of 0.8 in order to identify and discard sampling circles that fall on the image background.

The proposed vectorisation algorithm identifies 306 of the 310 junctions present in these test drawings, with a mean localisation error of  $4.7 \pm 2.3$  pixels, resulting in straight line vectors which are well placed with respect to the drawn edges. The algorithm fails to identify two junctions in drawing (f) and another junction in drawings (g) and (k). This is due to the fact that in these drawings, the length of the segments at the junction are smaller than the size of the circle samplers, lowering the co-occurrence value of these lines since they do not co-occur across all circles in the family of concentric circle samplers.

The performance of the proposed vectorisation algorithm was compared with that obtained by the Sparse Pixel Vectorisation (SPV) algorithm described in [2]. The performance was measured using the pixel recovery index (PRI) proposed in the vectorisation evaluation protocol of [11]. This measures the degree of overlap between the detected vectors and the ground-truth representation of the drawings and consists of two components, namely the true detection rate  $D_p$  and the false detection rate  $F_p$  such that  $PRI = 0.5D_p + 0.5(1 - F_p)$ . Ideally, the true detection rate  $D_p = 1$  while the false detection rate  $F_p = 0$  such that  $PRI = 1$ . The results obtained are given in Table 1. Here we can note that the SPV algorithm has high  $PRI$  values for drawings (a) - (j), as expected from the results reported in [2]. We can also observe that the performance of the proposed circle-based vectorisation is at an equal level as the SPV algorithm for these drawings. Performance differs for Drawing (k), where, due to the required binarisation of the drawing, the SPV also vectorises the faint lines of the paper background, resulting in a large number of false detections, since these lines do not form part of the sketched drawing. Using the proposed circle-based vectorisation algorithm, these lines do not provide a significant accumulation value in the co-occurrence matrix  $S(\mathbf{d}, \theta, \Delta\theta)$  and thus are not represented as vectors, such that only vectors representing the sketched drawing are retained by the proposed algorithm, as desired. Moreover, the proposed circle-vectorisation algorithm can identify and label the junction points while obtaining the vector representation of the drawing.

## 6 Conclusion

In this paper, we present the hypothesis that junction points may be localised within some sampling circle by noting the co-occurrences of the intersecting angles between the circle samples and the line strokes that form the junction. Hence, we show that the drawing can be vectorised using multiple circles without the need for path following, selection of suitable located seed pixels or image binarisation. The algorithm detected 98% of the junctions from the drawings on which it was evaluated.

**Table 1.** Comparison of the performance of the circle-based vectorisation algorithm with the sparse pixel vectorisation algorithm, showing the pixel detection rate,  $D_p$ , the false detection rate  $F_p$  and the pixel recovery index  $PRI$ .

Drawing	SPV			Circle-based vectorisation		
	$D_p$	$F_p$	$PRI$	$D_p$	$F_p$	$PRI$
(a)	0.90	0.06	<b>0.92</b>	0.91	0.11	<b>0.90</b>
(b)	0.89	0.06	<b>0.92</b>	0.86	0.10	<b>0.88</b>
(c)	0.91	0.05	<b>0.93</b>	0.92	0.09	<b>0.91</b>
(d)	0.88	0.09	<b>0.90</b>	0.91	0.11	<b>0.90</b>
(e)	0.90	0.13	<b>0.89</b>	0.92	0.09	<b>0.91</b>
(f)	0.86	0.11	<b>0.87</b>	0.78	0.10	<b>0.84</b>
(g)	0.89	0.06	<b>0.92</b>	0.86	0.10	<b>0.88</b>
(h)	0.85	0.11	<b>0.87</b>	0.90	0.11	<b>0.89</b>
(i)	0.88	0.11	<b>0.89</b>	0.87	0.11	<b>0.88</b>
(j)	0.88	0.08	<b>0.90</b>	0.82	0.20	<b>0.81</b>
(k)	0.72	0.75	<b>0.48</b>	0.77	0.10	<b>0.84</b>

The algorithm requires that sampling circles are placed on a dense grid in order to ensure that each junction point appears in at least one sampling circle. While this ensures that all junctions can be sampled, the circle samplers may occasionally fail to locate a junction point. This is mainly due to the fixed size of the sampling circles while the drawing may contain regions of mixed resolution. Thus, the vectorisation may be further improved by allowing the size of the family of concentric sampling circles to adjust according to the local drawing detail resolution.

## References

1. Bonnici, A., Camilleri, K.: A circle-based vectorization algorithm for drawings with shadows. In: [Proceedings of the International Symposium on Sketch-Based Interfaces and Modelling](#). pp. 69–77 (2013)
2. Dori, D., Wenyn, L.: Automated cad conversion with the machine drawing understanding system: concepts, algorithms, and performance. *Systems, Man and Cybernetics, Part A: Systems and Humans*, IEEE Transactions on 29(4), 411–416 (1999)
3. Gennari, L., Kara, L.B., Stahovich, T.F., Shimada, K.: [Combining geometry and domain knowledge to interpret hand-drawn diagrams](#). *Computers & Graphics* 29(4), 547 – 562 (2005)
4. Guerreiro, R.F.C., Aguiar, P.M.Q.: [Extraction of line segments in cluttered images via multiscale edges](#). In: [Proceedings of the 2013 IEEE International Conference of Image Processing](#) (2013)
5. Hasson, N., Aljunid, S., Ahmad, R.: [Extract dominant elements and shapes from raster images](#). In: [Electronic Design, 2008. ICED 2008. International Conference on](#). pp. 1–4 (2008)

6. Hilaire, X., Tombre, K.: Robust and accurate vectorization of line drawings. [IEEE Transactions on Pattern Analysis and Machine Interpretation 28\(6\), 890–904 \(2006\)](#)
7. Jin, W., Hujun, B., Weihua, Z., Qunsheng, P., Yingqing, X.: Automatic image-based pencil sketch rendering. [J. Comput. Sci. Technol. 17\(3\), 347–355 \(May 2002\)](#)
8. Kang, S., Choung, Y., Park, J.: Image corner detection using hough transform. In: Marques, J., Prez de la Blanca, N., Pina, P. (eds.) *Pattern Recognition and Image Analysis, Lecture Notes in Computer Science*, vol. 3523, pp. 279–286. Springer Berlin Heidelberg (2005)
9. Katz, R.A., Pizer, S.M.: Untangling the blum medial axis transform. [International Journal of Computer Vision 55\(2-3\), 139–153 \(November 2004\)](#)
10. Liu, J., Chen, Y., Tang, X.: Decomposition of complex line drawings with hidden lines for 3d planar-faced manifold object reconstruction. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33(1), 3–15 (2011)
11. Liu, W., Dori, D.: A protocol for Performance Evaluation of Line Detection Algorithms. *Machine Vision Applications* 9, 240 – 250 (1997)
12. Liu, W., Dori, D.: Sparse pixel vectorisation: An algorithm and its performance evaluation. [IEEE Transactions of Pattern Analysis and Machine Intelligence 21\(3\), 202–215 \(1999\)](#)
13. Lu, T., Yang, Y., Yang, R., Cai, S.: Knowledge extraction from structured engineering drawings. In: *Fuzzy Systems and Knowledge Discovery, 2008. FSKD '08. Fifth International Conference on*. vol. 2, pp. 415–419 (2008)
14. Myers, R., Hancock, E.R.: Genetic algorithms for ambiguous labelling problems. [Pattern Recognition 33\(4\), 685 – 704 \(2000\)](#)
15. Nagy, G.: Twenty years of document image analysis in pami. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 22(1), 38–62 (2000)
16. Naouai, M., Narjess, M., Hamouda, A.: Line extraction algorithm based on image vectorization. In: *Mechatronics and Automation (ICMA), 2010 International Conference on*. pp. 470–476 (2010)
17. Nidelea, M., Alexei, A.: Method of the square - a new algorithm for image vectorization. In: *Communications (COMM), 2012 9th International Conference on*. pp. 115–118 (2012)
18. Song, J., Su, F., Chen, J., Tai, C., Cai, S.: An object-oriented progressive-simplification-based vectorization system for engineering drawings: Model, algorithm, and performance. [IEEE Transactions of Pattern Analysis and Machine Intelligence 24\(8\) \(2002\)](#)
19. Song, J., Lyu, M.R.: A hough transform based line recognition method utilizing both parameter space and image space. [Pattern Recognition 38\(4\), 539–552 \(2005\)](#)
20. Sousa, M.C., Buchanan, J.W.: Observational models of graphite pencil materials. [Computer Graphics Forum 19\(1\), 27–49 \(2000\)](#)
21. Tombre, K., Ah-Soon, C., Dosch, P., Masini, G., Tabbone, S.: Stable and robust vectorization: How to make the right choices. *Graphics Recognition - Recent Advances, GREC'99, Lecture Notes in Computer Science*, Springer-Verlag 1941, 3–18 (2000)
22. Wei, J., Li, M., Wang, Y., Chen, C., Hong, W., Chen, Z.: Parallel algorithm designed for polygon vectorization. In: *Geoinformatics (GEOINFORMATICS), 2012 20th International Conference on*. pp. 1–4 (2012)