# Convex hulls, occluding contours, aspect graphs and the Hough transform [1]

M. Wright    A. Fitzgibbon    R. B. Fisher

Department of Artificial Intelligence

University of Edinburgh

Edinburgh, Scotland EH1 2QL

markwr, andrewfg or rbf@aifh.ed.ac.uk

### Abstract

The Hough transform is a standard technique for finding features such as lines in images. Typically edgels or other features are mapped into a partitioned parameter or Hough space as individual votes. The target image features are detected as peaks in the Hough space. In this paper we consider not just the peaks but the mapping of the entire shape boundary from image space to the Hough parameter space. We analyse this mapping and illustrate correspondences between features in Hough space and image space. Using this knowledge we present an algorithm to construct convex hulls of arbitrary 2D shapes with smooth and polygonal boundaries as well as isolated point sets. We also demonstrate its extension to the 3D case. We then show how this mapping changes as we move the origin in image space. The origin can be considered as a vantage point from which to view the object and the occluding contour can be extracted easily from Hough space as those points where $R = 0$. We demonstrate the potential for tracking of transitions in the mapping to be used to construct an aspect graph of arbitrary 2D and 3D shapes.

## 1    Introduction

The Hough transform [2] is a standard technique in computer vision having many applications for finding lines, circles and other features in images [3]. Typically edgels or other features are mapped into a partitioned parameter or Hough space as individual votes. The target image features are detected as peaks in the Hough space.

One particular variant of the Hough transform for line detection is shown in figure 1. This uses the $R\theta$ parameterization and makes use of edgel orientation to reduce computation [4]. For each edgel a perpendicular is dropped from the origin in image space to a line passing through the edgel and with the same orientation. This is then mapped to a space with two parameters $R$ and $\theta$ where $R$ is the length of perpendicular and $\theta$ is the angle it makes with the $x$axis. As orientation information is avialable then each edgel maps to a single point in Hough space. If the edgels lie on a line then all will map to a common point in the Hough space
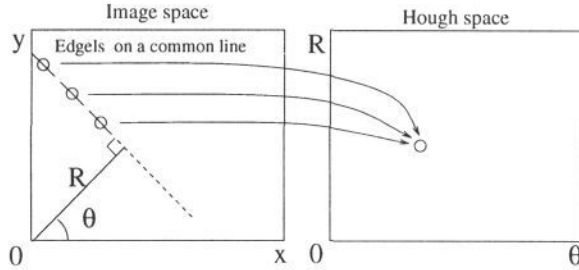
---

Figure 1: On the left is a set of edgels in image space with the same orientation and aligned such that they lie of the same line. On the right we can see that they map to a single point in Hough space and so the line they create can be detected as a peak at this point.

and this peak in the Hough space or accumulator signifies a linear feature in image space. If no orientation information is available then each edge point maps to a sine wave in Hough space corresponding to the set of all lines passing through that point. The intersection of these sine waves signifies the parameters of the line that passes through all the points. In this paper we consider not just the peaks in Hough space but the mapping of a whole object boundary from image space to $R, \theta$ space. The layout of the paper is as follows:

Firstly we examine this mapping and find that for a smooth object boundary this is a continuous curve. We outline what features of the Hough space curve relate to features of the object boundary. We then show how the Hough transform can be used to generate the convex hull of an object if a stationary origin is used in image space. We also show how the problem of discontinuity can be overcome and demonstrate the extension to 3D.

We then consider a dynamic situation where the origin is allowed to move anywhere. We show that this can be used to extract the apparent contour and tracking of features in Hough space could be the basis for an algorithm to compute aspect graphs for arbitrary 2D and 3D shapes.

## 2   The $R, \theta$ mapping of the shape boundary

In the Hough transform algorithm described above individual edgels are mapped into the Hough space and the peaks detected. Those edgels not contributing to peaks in the accumulator are ignored. We will now consider what happens to these other edgels and what this can tell us about the geometry of the curve. Consider the smooth image curve in figure 2a. We can sample the boundary of this curve and at each point determine the orientation of the tangent to the curve. As we traverse the curve in image space a curve is also traced out in $R, \theta$ space. The mapping to Hough space for our smooth curve is given in figure 2b. This curve is remarkably structured and we will now describe how its geometric features relate to the image space curve.

The main features of the curve are self intersections, cusps, maxima, minima and inflexions. The self intersections of the $R, \theta$ curve at $C$ and $F$ in figure 2b
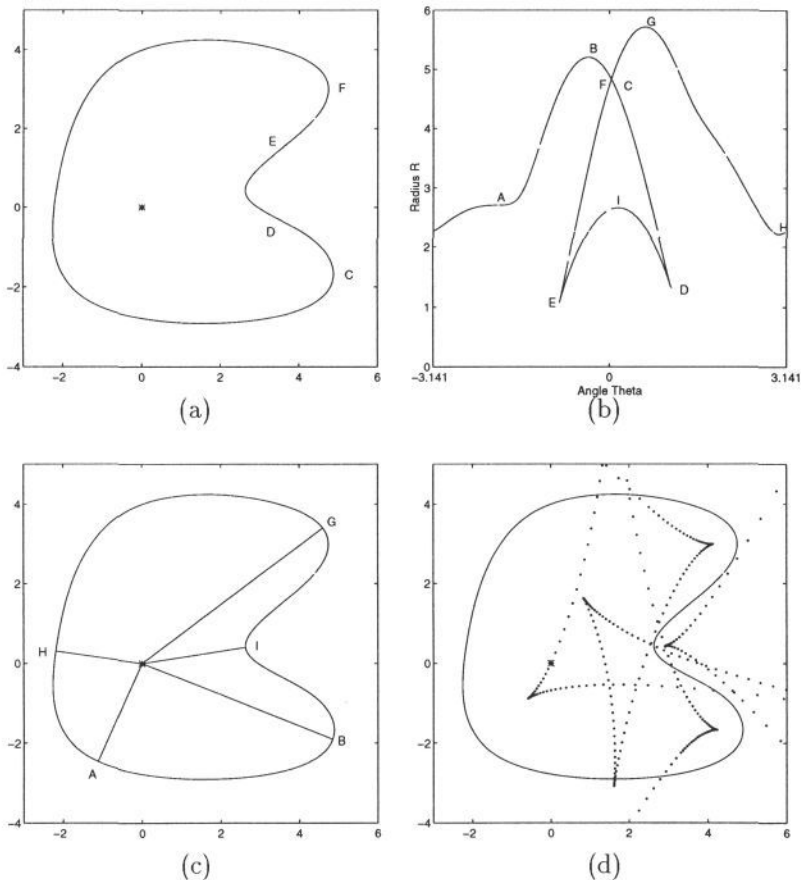
Figure 2a shows a smooth image curve. Figure 2b shows the mapping of the image curve to $R, \theta$ or Hough space. Features on the Hough space curve have been labeled to show corresponding features in image space depicted in figures 2a and c (see text). Figure 2d shows the evolute of the image curve. The relationship between the evolute (dotted line) and origin (astrix) determine the type of turning points that exist on the Hough space curve (see text).

Figure 2: The mapping from image space to $R, \theta$ space

are due to two points on the curve boundary at $C$ and $F$ in figure 2a mapping to the same point in Hough space. This signifies where there is a bi-tangent line in the image *i.e.* a line tangent to the boundary at two distinct points. The loop formed by the self intersection corresponds in this case to the indentation in the curve boundary. The cusps within this loop at $D$ and $E$ in figure 2b correspond to points of inflexion at $D$ and $E$ on the image space curve in figure 2a. The maxima $(B, G, I)$, minima $(H)$ and inflexion $(A)$ points on the Hough space curve in figure 2a all correspond to points on the shape boundary where the normal at

that point passes through the origin as shown in figure 2c. It is a minima if the distance along the the normal measured from the shape boundary to the origin is less than the radius of curvature of the curve at that point. It is a maxima if the normal length is greater than the radius of curvature and it is an inflexion if they are equal. The locus of centres of curvature defines a cusped curve called the evolute which is depicted over the shape boundary as a dotted line in figure 2d. So the minima correspond to an origin within the evolute, maxima have an origin beyond the evolute and inflexions occur when the origin is on the evolute.

# 3   The $R, \theta$ mapping and convex hull

The relationship between the $R, \theta$ mapping and the convex hull is as follows. If we take for every value of $\theta$ that portion of the Hough space curve that has maximum $R$ *i.e.* the upper envelope of the $R, \theta$ space graph. Then the corresponding points in image space are those points on the convex hull.

## 3.1   Dealing with discontinuity

It would appear at first sight that this might only work for smooth curves because if the derivative of the boundary is discontinuous then there is a break in the Hough space curve. It is however a simple matter to join up these gaps. We remember that, for the original Hough transform, if there is no orientation information then points in image space map to sine waves in Hough space. At a corner we merely join the end of one edge to the beginning of the next with the appropriate sine wave. In fact this idea can be taken to its extreme if we wish to construct the convex hull of sets of isolated points. In figure 3a we see a point set in image space. Each of these points maps to a sine wave in Hough space which are labeled in figure 3b. The convex hull consists of all those points for which their corresponding Hough space sine wave is maximal in $R$ for any particular value of $\theta$. Further more their ordering around the hull is given simply by their adjacency in Hough space. Only sine waves $1, 2$ and $3$ have a maximal $R$ and are on the convex hull. Sine wave $5$ never approaches a maximal position but wave $4$ comes close at the intersection of wave $1$ and $2$. This is because point $4$ in image space is nearly collinear with points $1$ and $2$. In figure 4 we see the convex hull of 7870 isolated points randomly distributed within a unit circle.

## 3.2   Extending to 3D hulls

Extending the approach to 3D is straight forward. We now drop a perpendicular to a tangent plane rather than a tangent line. The parameters are the distance $R$ of the plane from the origin and $\theta_1$ and $\theta_2$ which are the azimuth and elevation of the perpendicular. The Hough space now has three dimensions rather than two. (In the general case we drop the perpendicular to a hyperplane and the Hough space becomes n-dimensional.) Figure 5 shows a 3D point set with the points on the convex hull marked with circles. Figure 6 shows the Hough space for this three dimensional point set. This is the upper envelope of the sinusoids generated by all points and each one corresponds to a point on the convex hull.
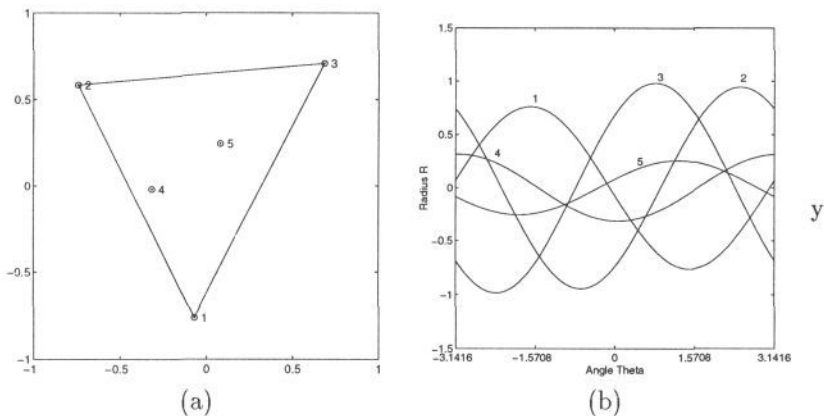
(a)  (b)

Figure 3a shows a set of 5 image points and their convex hull. In figure 3b we can see their corresponding sine waves in Hough space. Only those waves with maximal $R$ for some value of $\theta$ contribute to the convex hull, in this case waves $1, 2$ and $3$.

Figure 3: The relationship between the convex hull and the Hough transform for isolated points
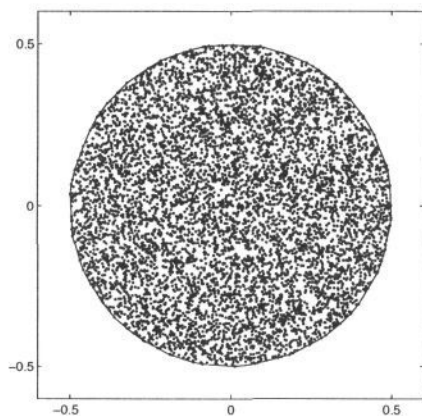


Figure 4: The convex hull of 7870 points generated using the Hough transform
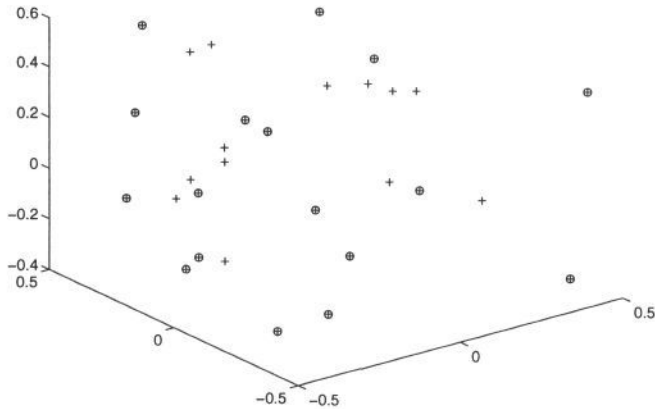
Figure 5: A three dimensional point set. Points on the convex hull are ringed with circles.
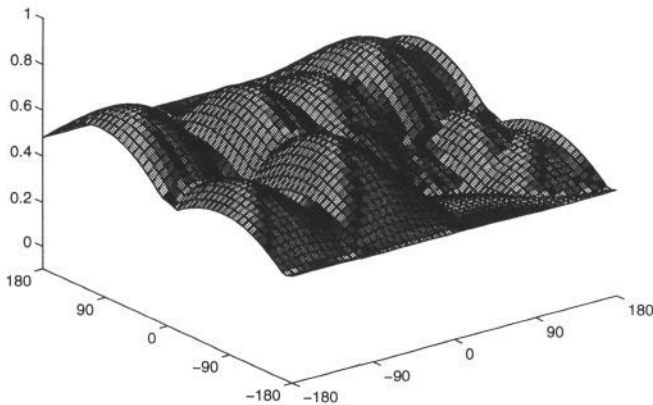


Figure 6: The Hough space for the set of 3D points in figure 5. The surface is of the upper sinusoids in Hough space corresponding to points on the convex hull.

## 3.3 Algorithms

### 3.3.1 Algorithm 1: sampling

In Hough space the upper envelope of all sine waves defines the convex hull, in that each sine wave segment on this envelope corresponds to a single point in the hull. We now describe an algorithm to identify these segments. The array *hull* is workspace.

**Step 1** Let $p_1(\theta), \ldots, p_n(\theta)$ be the sine waves in Hough space corresponding to the $n$ points in the image space.

**Step 2** Set index $k$ to 1; for $\theta = 0$ to $2\pi$ in steps of $\Delta_\theta$ (see discussion below) fill an array $hull[k]$ with indicies as follows:

**Step 2a** Find the greatest sine at this $\theta$: Set $hull[k] := i$ where $p_i(\theta) \geq p_j(\theta)$ for all $1 \leq j \leq n$.

**Step 2b** Increment index $k$ by 1.

**Step 3** The distinct indicies in $hull$ are those points in the image plane in the convex hull.

(Termination is trivial since $\Delta_\theta > 0$.)

An important issue is the granularity of the Hough space from the viewpoint of the complexity of computing the convex hull by this algorithm. Consider three collinear points $a$, $b$ and $c$, with $a$ and $b$ in the convex hull, and $c$ lying between them. Without loss of generality we assume that $c$ lies at the origin of the image plane and $a$ and $b$ on either side of it along the $x$-axis, at $-x_a$ and $x_b$. We will consider the behaviour in Hough space as $c$ is perturbed along the $y$-axis by $\delta c$, and derive from this the required sampling interval of the Hough space so as to detect that $c$ is now part of the hull.

Perturbing $c$ in this way contributes $\delta c . \sin \theta$ to Hough space, and we see that the interval $\Delta_\theta$ must be such that

$$\Delta_\theta < \tan^{-1}(\delta c/x_a) + \tan^{-1}(\delta c/x_b)$$

for $c$ to be seen to be in the hull. This gives an approximate complexity $O(\frac{n}{\tan^{-1}(1/p)})$ where $p$ measures the greatest distance between any two points $a$ and $b$. Choosing $p$ a priori to be the image diagonal renders it independant from (a bounded) $n$, and overall complexity is $O(n)$; however, constants are much greater than in the Graham's scan algorithm [1] [6].

For example, for a $512 \times 512$ image we have $512.4\pi n$ for $n$ points with a twice-over sampling interval.

### 3.3.2 Algorithm 2: non-sampling

We can make a significant (see below) average-case improvement in speed of the algorithm, with the following approach, which has complexity $O(n^2)$. (We adopt the same notation as above.)

**Step 1** Determine $i$ such that $p_i(0) > p_j(0)$ for all $1 \leq j \leq n$. We use this hull point to start.

**Step 2** Record that the point $i$ is on the hull. If $i = 2\pi$ then we have completed the hull, so stop.

**Step 3** Compute the next point $i'$ known to be on the hull as follows. Determine the smallest $\theta_{i'}$ such that $p_i(\theta_{i'}) = p_{i'}(\theta_{i'})$ subject to $\theta_{i'} > \theta_i$. $i'$ is the 'nearest' sine wave which intersects with since wave $i$.

Set $i$ to $'i$ and go to Step 2.

(We omit proofs of termination.)

This algorithm has quadratic complexity since there are $n$ computations in Step 3 (finding the next interection point), and we do this for each point in the hull, which is $O(n)$ (think of all $n$ points arranged as as circle). Hence $O(n^2)$.

However, we anticipate much better that this worst-case performace in practice, since typically considerably fewer points are in the hull in comparision with $n$.

# 4   The $R, \theta$ mapping and aspect graph

In the last section we computed the convex hull of 2D and 3D objects by using the $R, \theta$ mapping of object boundaries with a stationary origin in image space. In this section we consider what happens to the same mapping as we move the origin in image space.

If the origin is outside the shape then we can consider it as a vantage point from which we look at the object. Using this perspective the first striking observation is that the occluding contour of the object, *i.e.* those point where the local tangent line or plane lies along the line of sight, is simply those points where $R = 0$. Further more those points with $R > 0$ have normals pointing toward the origin or vantage point and those with $R < 0$ have normals points away. We suggest that these observations can form the basis of an algorithm to build aspect graphs [5] of arbitrary 2D and 3D shapes. We can move the origin about the viewing sphere and keep track of the self intersections and cusps of the Hough space curve. As these features cross the $\theta$ axis *i.e.* $R = 0$ these constitute event boundaries in the aspect graph. To illustrate this principle we have devised a graphics program which animates in real time the mapping from image space to Hough space as the origin is moved or the shape boundary is deformed. Figure 7 shows a few different frames of the animation. As the vantage point (cross) is moved so the concavity comes into view the self intersection and cusps of the curve in Hough space move below the $\theta$ axis. We suggest that tracking of these events could be used to trigger the building of new branches and nodes in the aspect graph.

# 5   Discussion

The convex hull algorithms presented above has similarities to Graham's algorithm where points are sorted according to angle. The complexity of Graham's algorithm is $O(n \ln(n))$. However our Hough based approach (algorithm 1) has a high constant term due to the sampling of Hough space. Algorithm 2 has much poorer worse case complexity but in the average case we anticipate better performance than algorithm 1. Neither algorithm can compete with Graham's algorithm in performance as a practical convex hull algorithm. However, our goal was not to produce an efficient new algorithm but to investigate the relationship between the convex hull and the Hough transform.

We believe that sampling may also be the basis for producing approximate convex hulls (and aspect graphs) in a principled way which has been identified as an important problem. Unlike Graham's algorithm there is a global representation of the hull. We could in principle use this to assess the effect of leaving out a

(a)  (b)  (c)  (d)  (e)  (f)
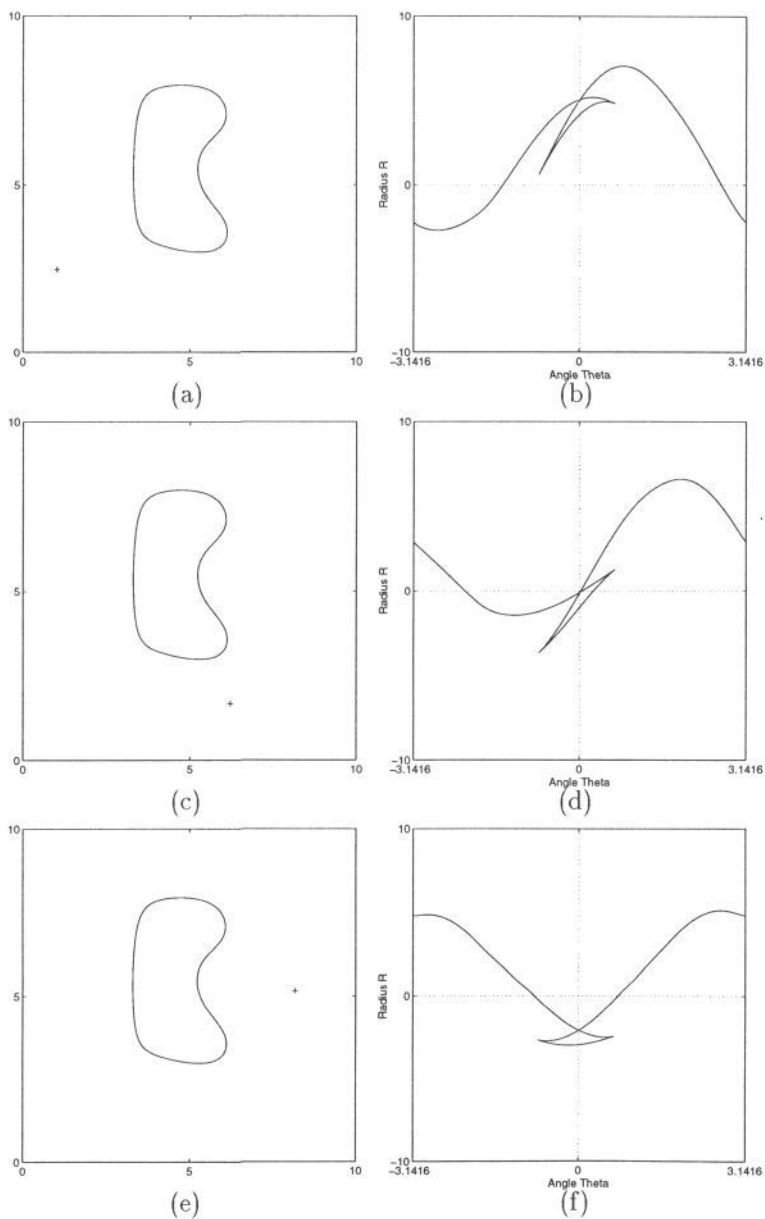
Figures 7a,c and e show a kidney bean shape in image space where different view points are marked with a cross. In figure 7a no part of the concavity can be seen and the loop in Hough space in figure 7b is above the $x$ axis. In figure 7c the viewpoint is precisely along the bi-tangent line, this is signified by the self intersection in figure 7d crossing the $x$ axis. In figure 7e the concavity is in plain view and the loop in figure 7f is below the $x$ axis.

Figure 7: The transitions in Hough space relating to changing view point.

particular point. This could be on the principle of the area of the sine wave for which it is maximal in $R$.

The Hough approach to building the aspect graph has the problem that although we can tell if a face points towards the viewer we cannot tell if it is occluded by another part of the object. It is as if we are considering the singularities of a transparent object. We could solve this by filtering the Hough space output with some kind of visibility check. One such check would be to compute a different transform, say $d, \alpha$ where $d$ is the length of a line from the boundary point to the origin and $\alpha$ is the angle this line makes with the $x$ axis. The points in $d, \alpha$ space with minimal $d$ for a particular $\alpha$ are those points which are unoccluded.

# 6   Conclusions

We have considered the mapping of object boundaries using the Hough transforms $R, \theta$ parameterization. We have shown the correspondence of features in Hough space to image space. This information has been used to construct an algorithm to compute 2D and 3D convex hulls. We considered the effect of moving the origin and demonstrated the potential of the approach to compute aspect graphs of arbitrary shapes. We hope that we have challenged the traditional role of the Hough transform an purely a feature detector. Clearly the local structure and dynamics of this mapping can tell us alot about the geometry of shapes around us.

# References

[1] R. L. Graham. An efficient algorithm for determining the convex hull of a finite planar set. *Information processing letters*, 1:132–133, 1972.

[2] P. V. C. Hough. *Method and means for recognising complex patterns*. US Patent 3069654, 1962.

[3] J. Illingworth and J. Kittler. A survey of the Hough transform. *CVGIP*, 44:87–116, 1988.

[4] C. Kimme, D. Ballard, and J. Sklansky. Finding circles by an array of accululators. *Comm ACM*, 18:120–122, 1975.

[5] J. J. Koenderink and A. J. van Doorn. The internal representations of solid shape with respect to vision. *Biological Cybernetics*, 32:211–216, 1979.

[6] J. O'Rourke. *Computational geometry in C*. Cambridge University press, 1992.