Figure-Ground Organization Based on 3D Symmetry

Aaron Michaux[a,*], **Vijai Jayadevan**[a], **Edward Delp**[a], **Zygmunt Pizlo**[a,b]
[a]Purdue University, School of Electrical and Computer Engineering, 465 Northwestern Ave, West Lafayette, USA, 47907
[b]Purdue University, Department of Psychology Sciences, 703 3rd Street, West Lafayette, USA, 47907

Abstract. We present an approach to figure/ground organization using mirror symmetry as a general purpose and biologically motivated prior. Psychophysical evidence suggests that the human visual system makes use of symmetry in producing 3D percepts of objects. 3D symmetry aids in scene organization because (i) almost all objects exhibit symmetry, and (ii) configurations of objects are not likely to be symmetric unless they share some additional relationship. No general purpose approach is known for solving 3D symmetry correspondence in 2D camera images, because few invariants exist. Therefore, we present a general purpose method for finding 3D symmetry correspondence by pairing the problem with the two-view geometry of the binocular correspondence problem. Mirror symmetry is a spatially global property that is not likely to be lost in the spatially local noise of binocular depth maps. We tested our approach on a corpus of 180 images collected indoors with a stereo camera system. K-means clustering was used as a baseline for comparison. The informative nature of the symmetry prior makes it possible to cluster data without a priori knowledge of which objects may appear in the scene, and without knowing how many objects there are in the scene.

Keywords: symmetry, clustering, figure/ground organization, biological vision.

*Aaron Michaux, amichaux@purdue.edu

1 Introduction

According to most studies of human vision the first step in visual perception is determining whether there are objects in front of the observer: where they are and how many there are. This step (visual function) is called *figure-ground organization* (FGO).[1] The computer vision community refers to this problem as object discovery. As with all natural visual functions of human observers, FGO operates in 3D space, as opposed to the 2D retinal image. It follows that it is natural to think about visual mechanisms underlying FGO as based on 3D operations. However, the fact that the input to the visual system is one or more 2D retinal images encouraged previous researchers to look for a theory of FGO based on 2D operations. This is how the human vision community studied FGO. Consider the prototypical example of Edgar Rubin's vase-faces stimulus.[2] In this 2D stimulus there are two possible interpretations depending on which region is perceived as a "figure" as opposed to the “ground”. Similar bistable stimuli have been used during the last several dozen years of FGO research in human vision.[3,4] This research provided a large body of results, but few theories and computational models. Furthermore, the proposed models are usually not suitable for real retinal or camera images representing 3D scenes. The present paper breaks with this tradition and looks for 3D operations that can establish the correct 3D FGO.

Once we assume that FGO refers to the 3D percept, we have to decide how the transition from the 2D image to the 3D percept is made. Here we follow the paradigm of inverse problems introduced to vision by Poggio et al.[5] According to this paradigm, inferences about 3D scenes, based on one or more 2D images, must involve a priori constraints (aka priors). Without constraints, the 3D inference problem is ill-posed, because there are infinitely many possible 3D interpretations that are consistent with any given 2D input. The a priori constraints are imposed on the family of possible interpretations resulting in a unique and accurate solution. We have already shown how this works with 3D shape perception.[6] Specifically, in 3D shape recovery, 3D symmetry is the natural prior. This makes sense because most, if not all, natural objects are symmetrical. There is empirical evidence showing that the human visual system relies on the 3D symmetry constraint.[7] The symmetry constraint is responsible for our veridical perception of 3D shapes. By “veridical,” we mean that we see shapes the way they are out there. In this paper we attempt to extend the operation of a 3D symmetry prior to FGO. Specifically, our theory of FGO is based on the following observation: almost all natural 3D objects are characterized by one or more types of symmetry, whereas a 3D configuration of unrelated objects is, itself, almost never symmetrical. In our theory the detection of symmetries in the 3D scene is equivalent to the detection of objects. There are as many objects in the scene as there are symmetries. Furthermore, the parameters of the symmetries (positions and orientations of the symmetry planes) provide information about the position and orientation of the objects in the 3D scene. The next section presents a brief overview of the symmetry prior.

1.1 The Generality of the Symmetry Prior

Assuming that objects are mirror symmetric may, at first, seem overly restrictive. Most real world scenes, however, are composed of 3D symmetric objects standing upright on a perceptually flat surface, such as the floor or simply the ground.[6,8]

Mirror-symmetric objects themselves tend to have a natural Cartesian coordinate system: front, back, left, right, top, and bottom. In a systematic treatment of spatial terms in language, Levinson referred to these types of object-centric directions as the *intrinsic reference frame*.[9] Cross-cultural linguistic analysis by Talmy further suggests that these object-centric directions are represented in *mentalese* – the native representation of mental information.[10] One of us argued that such a coordinate system exists as a consequence of purely physical properties of the world that we evolved in.[6] For example, it would be very hard for an animal to be physically stable and to move around if it were not bilaterally symmetric. DNA evidence in the field of molecular phylogenetics suggests that the first mirror symmetric organisms – the so called *bilateria* – evolved more than half a billion years ago,[11] and now constitute the vast majority of animal phyla, including the arthropods (e.g., insects and arachnids) and the chordates: animals with a hollow nerve chord running down their backs, e.g., sharks, birds, cats, fish, and humans. The few animals that are not bilateria, such as sponges and jellyfish, still show other forms of symmetry, such as radial symmetry. Symmetry is, therefore, a natural and general prior and it should be used in 3D vision. The symmetry plane is usually orthogonal to the ground because that provides the best support against gravity. The cross product of the ground-normal, up-down, (approximating the direction of gravity) and the normal of the symmetry plane, left-right, gives the third intrinsic direction: front-back.

Psychophysical evidence clearly shows that the human visual system makes use of the symmetry prior in 3D shape recovery.[7] Symmetry is also important in *shape constancy*. Shape constancy refers to the phenomenon where the perceived shape of an object is constant despite changes in the shape of the retinal image caused by changes in the 3D viewing direction. Experimental results on the role of symmetry in shape constancy and shape recovery in humans suggest that symmetry is an essential characteristic of shape.[8, 12]

1.2 Related research

Symmetry has already played an important role in computer vision research. This goes back at least to a landmark 1978 publication by Marr & Nishihara,[13] who emphasized the importance of 3D symmetrical shape parts based on Binford's[14] generalized cones. The presence of symmetry in a 3D object allows derivation of invariants of a 3D to 2D projection (for example Refs. 15–17). 3D symmetries also facilitate 3D recovery from a single 2D image using multiview geometry.[18] There have been some attempts to use 2D, as opposed to 3D symmetries in image segmentation,[19] and image understanding.[20] However, the use of 2D symmetries in computer vision faces fundamental difficulties simply because a 2D camera image of a 3D symmetrical object is, itself, almost never symmetrical.

Before a 3D symmetry prior is used to recover 3D shapes, 3D symmetry correspondence must be solved in the camera image, which itself is 2D and almost never symmetrical. Solving for symmetry correspondence has been tried for surfaces of revolution, which are characterized by rotational symmetry[21–23] as well as for mirror-symmetrical polyhedral objects, where edge features are compared with respect to 2D affine similarities (Refs. 24–26). The inherent difficulty of the

3D symmetry correspondence problem in 2D images has resulted in incremental successes, where the proposed methods work only for special cases such as nearly degenerate views (for example: Sinha, Ramnath & Szeliski[25]). The fact that the projection of a 3D mirror symmetric object into 2D rarely produces a symmetric image is only one part of the difficulty in solving for symmetry correspondence. An additional problem is that a camera image usually contains multiple objects. So, one must solve FGO before symmetry is applied to individual objects.

As pointed out in the beginning of this paper, FGO goes by the name of object discovery in the computer vision community. The state of the art of object discovery in real images makes extensive use of machine learning, and relies exclusively on 2D features. (For review, see Ref. 27.) The much harder problem of unsupervised object discovery has received comparatively little attention. In unsupervised object discovery, an algorithm analyzes an image in order to locate and label previously unseen objects. One approach is to discover the general characteristics of object categories from regularities in large sets of unlabeled training data. Those categories are then utilized to discover and locate objects in a set of testing images. For examples, see Refs. 28–32. This problem is typically considered so hard that most methods rely on at least some form of weak supervision. For example, Kim & Torralba[33] attempted to locate objects (Regions of Interest, ROI) without training data; however, a small set of initial exemplar ROIs must still be supplied.

While certainly acceptable in the computer vision community, the use of testing and training data is probably unimportant in human vision.[6] Human observers can detect and recover unfamiliar 3D shapes from a single 2D image, and recognize a 3D shape from a novel viewing direction.[6] Surely some learning can occur in human vision; an individual can learn and remember what an object looks like, but learning does not seem to be necessary for detecting 3D objects and recovering their shapes. Our approach aims at emulating what human observers do: our model does not use any training data, and there is no attempt to learn any category information, or regularities between exemplars. The present approach is not only unsupervised, it also uses an informative and generally applicable prior, 3D symmetry, in establishing FGO.

Object discovery is more effective when 3D points are available (for example, from stereo images). In such cases, a typical approach for object discovery is cluster analysis. K-means remains one of the most widely used clustering algorithms even though roughly fifty years have passed since it was independently discovered in various scientific fields. (For an historical review see Ref. 34.) Most clustering algorithms require, however, a priori knowledge of the number of clusters (i.e., objects in the scene), and these algorithms rely on some form of distance metric. Automatically determining the number of clusters is itself ill-posed, and often requires separate criteria for what is the "most meaningful" number of clusters. Specifying what is meaningful in a given application is a key problem which comes in addition to an over-reliance on uninformative priors, such as density and distance metrics. Our approach is to approximate the ill-posed clustering problem with a well-posed formulation based on 3D symmetry. Our algorithm uses 3D data from a binocular camera, however, it is the incorporation of a 3D symmetry prior that transforms clustering (object discovery) from an ill-posed problem into a well-posed one. The use of 3D symmetry can, at least in principle, lead to near perfect performance in FGO – the level of performance that characterizes human vision in everyday life.

2 Problem Formulation

There is no known way to achieve near-human performance in unsupervised object detection. Some definition of "object" and "background" is required for such an algorithm. In previous research, this definition is usually implicit in the clustering of low-level 2D features, such as interest

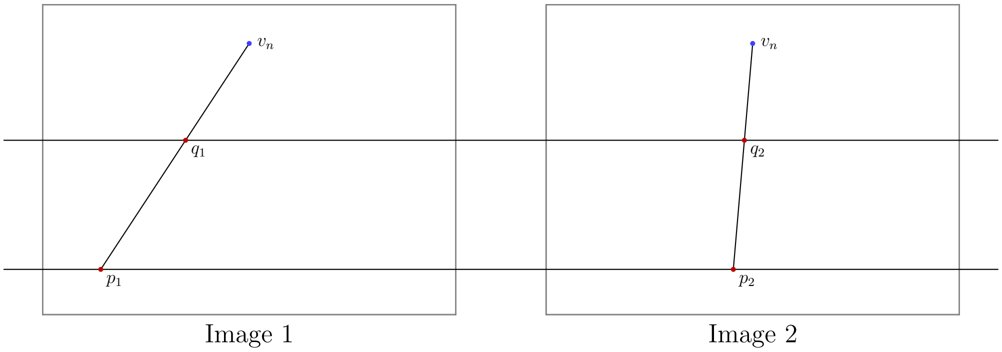

Fig 1: A *two-view mirror-symmetric quadruple* is a set of four 2D points that obey four constraints. Pairs of points $(\boldsymbol{p_1}, \boldsymbol{p_2})$ and $(\boldsymbol{q_1}, \boldsymbol{q_2})$ obey the epipolar constraint in definition 2.2.1: their *y*-coordinates are the same. Points $(\boldsymbol{p_1}, \boldsymbol{q_1})$ from image 1 and $(\boldsymbol{p_2}, \boldsymbol{q_2})$ from image 2 obey definition 2.3.1: they are colinear with the vanishing point. Note that the vanishing point is view-invariant across both images, and implied by the quadruple. It is not part of the quadruple.

points, in a training set. In this work we use the following psychologically motivated operational definition of an object in order to formulate the object detection problem:

Definition 2.1 An object's defining quality is its 3D shape, where shape is defined as a set of mirror symmetric curves with respect to a common symmetry plane.

Under this definition, polyhedral objects can be detected from a scene by locating one or more symmetry planes and then finding pairs of points that are mirror symmetric with respect to a particular symmetry plane. There is no known general purpose algorithm for finding 3D mirror symmetric curves in single 2D camera images; however, it is possible to simplify the symmetry correspondence problem by pairing it with the binocular correspondence problem, because the epipolar geometry of each provide non-overlapping constraints on the solution space. In effect, we simultaneously disambiguate each correspondence problem by using the epipolar geometry of the other.

We further simplify the problem by assuming that the objects' symmetry planes are orthogonal to the ground. This is typical for most objects which must resist gravity when standing on a flat surface. This additional assumption can, however, be removed without changing how the algorithm works.

2.1 *Notation*

We use upper-case bold letters, e.g., $\boldsymbol{X}$, to denote the coordinates of 3D points. Lower-case bold letters, e.g., $\boldsymbol{x}$, denote the projection of a 3D point onto the image plane of a camera. Subscripts are added to lower-case bold letters to identify a particular camera. For example, the projection of $\boldsymbol{X}$ in the first camera is $\boldsymbol{x_1}$, and in the second camera, $\boldsymbol{x_2}$. A star superscript is used to denote the homogeneous versions of these points. Therefore $\boldsymbol{X}^* \in \mathbb{P}^3$ is the homogeneous representation of the 3D point $\boldsymbol{X}$, and $\boldsymbol{x}^* \in \mathbb{P}^2$ is the homogeneous representation of the 2D point $\boldsymbol{x}$.

2.2 *Stereo Correspondence*

We use a pinhole camera model for each of a pair of calibrated cameras with identical intrinsic parameters, and with neither skew nor radial distortion. The center of camera 1 is located at the origin of the world coordinate system, $\boldsymbol{C}_1 = (0, 0, 0)^\top$, and the center of camera 2 lies on the x-axis at $\boldsymbol{C}_2 = (\delta_x, 0, 0)^\top$, where δ_x is the distance between the two cameras. The principal rays of the cameras are parallel and pointing down the negative z-axis. Stereo rectification is unnecessary under these assumptions, and we can define the epipolar geometry for the two-view camera system according to the following definition:

Definition 2.2.1 Let the ideal point $\boldsymbol{e_1}^* = (1, 0, 0)$ be the image of $\boldsymbol{C}_2$ in camera 1, and the ideal point $\boldsymbol{e_2}^* = (-1, 0, 0)$ be the image of $\boldsymbol{C}_1$ in camera 2. Then the corresponding lines between the two images are the horizontal scan-lines with identical *y*-coordinates. Thus for all 3D points $\boldsymbol{Z}$ we have $\boldsymbol{z_1}_y = \boldsymbol{z_2}_y$.

2.3 Symmetry Correspondence

A pair of 3D points, $\boldsymbol{P}$ and $\boldsymbol{Q}$ are mirror symmetric about a *plane of symmetry*, $\boldsymbol{\pi} = (n_x, n_y, n_z, d)^\top = (\boldsymbol{n}^\top, d)^\top$, if the plane of symmetry bisects a line segment connecting these two points. This, in turn implies the following two equations: (1), the two points are equidistant from the symmetry plane, and (2), the line joining the two points is parallel to the normal of the symmetry plane.

$$\frac{\boldsymbol{P}+\boldsymbol{Q}}{2} \cdot \boldsymbol{n} + d = 0 \tag{1}$$

$$(\boldsymbol{P}-\boldsymbol{Q}) \times \boldsymbol{n} = \boldsymbol{0} \tag{2}$$

We refer to the normal of the symmetry plane, $\boldsymbol{n}$, as the *direction of symmetry*. Without loss of generality assume that $\boldsymbol{n}$ is a unit normal. In this case d is scaled to the units of the coordinate system.

The plane of symmetry $\boldsymbol{\pi}$ defines its own epipolar geometry as given in definition 2.3.1.

Definition 2.3.1 The vanishing point of the 3D symmetry lines is isomorphic with the direction of symmetry. Let $\boldsymbol{n}$ be a direction of symmetry for an object. When extended to infinity, the projection of all 3D lines parallel to $\boldsymbol{n}$ meet at a vanishing point $\boldsymbol{v_n}$.

The vanishing point in definition 2.3.1 is commonly referred to as the epipole of the symmetry plane, and the lines passing through the epipole as the epipolar lines. To avoid a conflict of terminology between the binocular and mirror-symmetric epipolar geometries, we simply refer to the symmetry plane's epipole as the vanishing point, and the epipolar lines as the pencil of lines through the vanishing point. This pencil of lines is used to constrain symmetry correspondence in a single image, as described in property 2.3.1.

Property 2.3.1 If $\boldsymbol{p}$ and $\boldsymbol{q}$ are images of 3D points $\boldsymbol{P}$ and $\boldsymbol{Q}$ symmetric about $\boldsymbol{\pi} = (\boldsymbol{n}^\top, d)^\top$ with direction of symmetry $\boldsymbol{n}$, then $\boldsymbol{p}$ and $\boldsymbol{q}$ are colinear with the vanishing point $\boldsymbol{v_n}$, which is defined by the direction of symmetry. See Ref. 35.

2.4 Combined Correspondences

Each of the two correspondence problems involves a pair of image points that must obey the geometrical constraints of the specified problem. In the binocular correspondence problem, according to definition 2.2.1, pairs of points, one from each image, must have the same y-coordinate. In the symmetry correspondence problem, according to definition 2.3.1, pairs of points from a single image must be colinear with the vanishing point $\boldsymbol{v_n}$ defined by the direction of symmetry. These two problems are combined by choosing two points, $\boldsymbol{p}_1$ and $\boldsymbol{q}_1$ from the first image, and two points $\boldsymbol{p}_2$ and $\boldsymbol{q}_2$ from the second image. A set of four such points is called a *two-view mirror-symmetric quadruple*, or simply quadruple for short. Figure 1 shows such a quadruple and the constraints that they obey. All objects under definition 2.1 are composed of these quadruples.

Note that because there is no rotation between the two cameras in our simplified two-view geometry, the direction of symmetry is identical with respect to both cameras, and thus so is the vanishing point.

2.5 3D reconstruction based on symmetry

Given a vanishing point $\boldsymbol{v_n}$, it is possible to reconstruct the 3D locations of points $\boldsymbol{P}$ and $\boldsymbol{Q}$ from their images $\boldsymbol{p}$ and $\boldsymbol{q}$. Figure 2 below shows the geometry of this situation. The vanishing point resides on the image plane and has 3D coordinates (v_x, v_y, f). Without loss of generality, assume that the camera centre, $\boldsymbol{C}$, is at the origin, $(0, 0, 0)$. In this case, (v_x, v_y, f) is, itself, the direction of symmetry. That means that the camera image of all the points on 3D rays parallel to (v_x, v_y, f) will form lines that intersect at $\boldsymbol{v_n}$.[36]

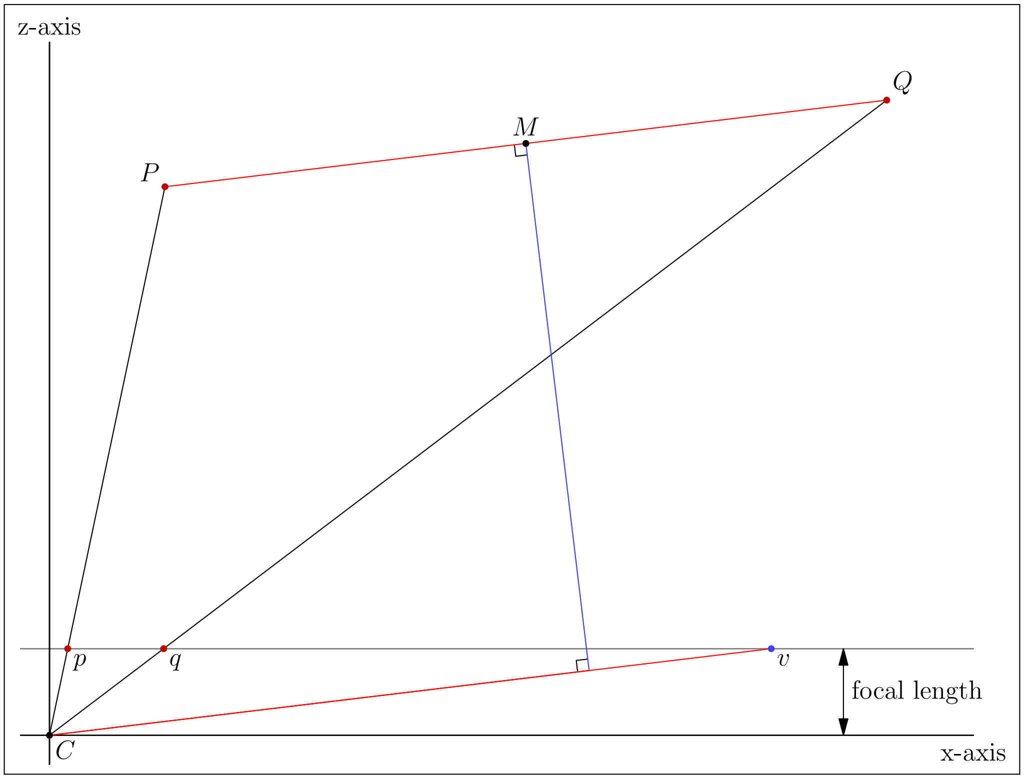

Fig 2: The imaged points $\boldsymbol{p}$ and $\boldsymbol{q}$, along with the vanishing point $\boldsymbol{v}$ define the location of 3D points $\boldsymbol{P}$ and $\boldsymbol{Q}$ up to scale. The symmetry plane can be written as $\boldsymbol{\pi} = (v_x, v_y, f, d)^\top$, where d locates the plane, and thus the point $\boldsymbol{M} = \frac{1}{2}(\boldsymbol{P} + \boldsymbol{Q})$ which lies on the plane. The vector $(\boldsymbol{P} - \boldsymbol{Q})$ is parallel to the direction of symmetry, creating a triangle that constrains the ratio of the lengths of the vectors $\|\boldsymbol{P}\|$ and $\|\boldsymbol{Q}\|$, allowing for 3D reconstruction, as specified by equations 3, 4, and 5.

The vector (v_x, v_y, f) is also the normal to the plane of symmetry that divides points $\boldsymbol{P}$ and $\boldsymbol{Q}$ at midpoint $\boldsymbol{M}$. Therefore the plane of symmetry can be written as $\boldsymbol{\pi} = (v_x, v_y, f, d)^\top$, where d is a free parameter that locates the plane of symmetry and scales the size of the reconstructed object.

By construction, the points $\boldsymbol{P}$ and $\boldsymbol{Q}$ must lie on rays emanating from the camera centre $\boldsymbol{C}$ through the imaged points $\boldsymbol{p}$ and $\boldsymbol{q}$. Given the intrinsic camera matrix K, let $\hat{\boldsymbol{p}} = \frac{K^{-1}\boldsymbol{p}^*}{\|K^{-1}\boldsymbol{p}^*\|}$, and $\hat{\boldsymbol{q}} = \frac{K^{-1}\boldsymbol{q}^*}{\|K^{-1}\boldsymbol{q}^*\|}$ be unit vectors in $\mathbb{R}^3$ that intersect the image plane at the desired points. We can then rewrite $\boldsymbol{P}$ and $\boldsymbol{Q}$ according to equation 3.

$$\boldsymbol{P} = \|\boldsymbol{P}\|\hat{\boldsymbol{p}} \quad \text{and} \quad \boldsymbol{Q} = \|\boldsymbol{Q}\|\hat{\boldsymbol{q}} \tag{3}$$

Let $\hat{\boldsymbol{v}} = \frac{K^{-1}\boldsymbol{v_n}^*}{\|K^{-1}\boldsymbol{v_n}^*\|}$ be the unit vector that intersects the image plane at the vanishing point $\boldsymbol{v_n}$. Now let $\theta = cos^{-1}(\hat{\boldsymbol{v}}^\top\hat{\boldsymbol{p}})$ be the angle between $\hat{\boldsymbol{v}}$ and $\hat{\boldsymbol{p}}$, and $\phi = cos^{-1}(\hat{\boldsymbol{v}}^\top\hat{\boldsymbol{q}})$ be the angle between $\hat{\boldsymbol{v}}$ and $\hat{\boldsymbol{q}}$. Then the ratio of $\|\boldsymbol{P}\|$ to $\|\boldsymbol{Q}\|$ is given by equation 4.

$$\frac{\|\boldsymbol{P}\|}{\|\boldsymbol{Q}\|} = \frac{sin\phi}{sin\theta} \tag{4}$$

This can be seen by construction. Referring to figure 2, let λ be the distance from the origin, to the line through $\boldsymbol{P}$ and $\boldsymbol{Q}$. Then by definition, $sin\theta = \frac{\lambda}{\|\boldsymbol{P}\|}$, and $sin\phi = \frac{\lambda}{\|\boldsymbol{Q}\|}$. Taking the ratio gives equation 4.

To solve for $\boldsymbol{P}$ and $\boldsymbol{Q}$, all that remains is to find the distance to one of the two points, and then substitute into equation 4 for the other. Note that $\boldsymbol{P}$ and $\boldsymbol{Q}$ are equidistant to the symmetry plane, giving $\boldsymbol{P}^\top\boldsymbol{v_n} + \boldsymbol{Q}^\top\boldsymbol{v_n} = 2d$. Substituting into equations 3 and 4 gives the equation:

$$\|\boldsymbol{P}\| = \frac{2d}{\hat{\boldsymbol{p}}^\top\boldsymbol{v_n} + \frac{sin\theta}{sin\phi}\hat{\boldsymbol{q}}^\top\boldsymbol{v_n}} \tag{5}$$

2.6 Using the Floor Prior

Every quadruple is consistent with a vanishing point as follows: find the point of intersection between the line going through $\boldsymbol{p}_1\ \boldsymbol{q}_1$, and the line going through $\boldsymbol{p}_2\ \boldsymbol{q}_2$. By definition, this point is colinear with both pairs of points, and therefore can be chosen provisionally as a vanishing point. Noting that, in homogeneous coordinates, the cross product of two points is the line going through them, and the cross product of two lines is the point of intersection, we have the equation for the vanishing point:

$$\boldsymbol{v}^* = (\boldsymbol{p}_1^* \times \boldsymbol{q}_1^*) \times (\boldsymbol{p}_2^* \times \boldsymbol{q}_2^*) \tag{6}$$

As shown in equation 5, a vanishing point and a pair of points from one image is sufficient for reconstruction. Although subject to more error from image quantization, triangulation[36] can also be used to locate 3D points $\boldsymbol{P}$ and $\boldsymbol{Q}$.

The floor prior can be used to define a restricted set of consistent quadruples. As previously pointed out, if symmetrical objects are standing on a flat surface, or floor, then the direction of symmetry will be in the 1-dimensional complementary subspace orthogonal to the floor normal. The projection of this subspace is the *horizon line* and it is isomorphic to the floor plane's normal.

With this prior, it is possible to determine the vanishing point for a pair of mirror symmetric points $\boldsymbol{p}$ and $\boldsymbol{q}$ in a single image. Let $\boldsymbol{g}$ be the normal to the floor, which is suggestive of the direction of gravity. $\boldsymbol{g}$ defines a line in homogeneous coordinates in the standard way (the intersection of the image plane with the plane parallel to the floor which passes through the camera center), and is called the horizon line. The vanishing point must be the point of intersection between the horizon line and the line passing through points $\boldsymbol{p}$ and $\boldsymbol{q}$.

$$\boldsymbol{v}^* = \boldsymbol{g} \times \boldsymbol{p}^* \times \boldsymbol{q}^* \tag{7}$$

Note, there is a degenerate case when $\boldsymbol{P}$, $\boldsymbol{Q}$, and the camera centre $\boldsymbol{C}$ lie on a plane parallel to the floor plane. In this situation $\boldsymbol{g} = \boldsymbol{p}^* \times \boldsymbol{q}^*$, and the vanishing point is underdetermined.

We can now determine if a two-view mirror-symmetric quadruple has a vanishing point that is consistent with an object standing on the floor. The estimate of $\boldsymbol{v}$ from equation 6 can entail substantial quantization error if pixels $\boldsymbol{p}_1$ and $\boldsymbol{q}_1$, or $\boldsymbol{p}_2$ and $\boldsymbol{q}_2$ are close to each other. For this reason, the vanishing point is estimated from the image where $\boldsymbol{p}$ and $\boldsymbol{q}$ are most distant, and a colinearity test is used to determine if the quadruple is consistent with the horizon defined by the floor normal.

Definition 2.6.1 A two-view mirror symmetric quadruple is consistent with the floor prior if there exists a vanishing point on the horizon line (defined by the normal to the floor) that is colinear both with $\boldsymbol{p_1}$, $\boldsymbol{q_1}$, and with $\boldsymbol{p_2}$, $\boldsymbol{q_2}$. A pair of points are considered colinear with the vanishing point if a line through the vanishing point exists such that the minimum point-line distance for both points is less than a threshold.

3 System Architecture

Input is a pair of stereo gray-scale images captured from a Point Grey Bumbleebee2® two-view camera system. A disparity map is calculated using the *Sum of Absolute Differences*, according to a propriety algorithm in the Triclops® 2.5 SDK, as published by Point Grey. The right image is used as the reference image for the disparity map.

The input images are smoothed with a Gaussian kernel before applying the canny operator with an adaptive threshold for hysteresis. The high threshold is automatically set to the average of the orientation magnitude at each pixel, as produced by the Sobel operators. The low threshold is automatically set to be 0.4 times the high threshold. A sparse disparity map is then produced by taking those disparity values from the disparity map which also register an edge in the edge maps for both images – within a pixel of error to account for quantization effects. The sparse disparity map gives corresponding pairs according to texture in the input image, but along identified edges in the image.

3.1 Estimating the Floor

Triangulation[36] is used to generate a point cloud from the disparity values. RANSAC is applied to find a 3D plane hypothesis that covers the maximal number of points, following a procedure outlined in Ref. 37. The normal to this plane is the estimate for $\boldsymbol{g}$. The floor points are then removed from analysis. The remaining sections below only consider non-floor points that are on identified edges in the image.

3.2 Finding Object Hypotheses

Two-view mirror-symmetric quadruples are defined as pairs of disparity map values from the sparse disparity map. A region of interest for symmetry correspondence is used to avoid searching through $O(n^2)$ pairs of disparity values. The set of floor-consistent quadruples, as per definition 2.6.1, are calculated from all pairs of disparity map values within the region of interest. All detected objects are subsets of these floor-consistent quadruples.

Segmentation of the scene proceeds by finding symmetry planes that produce spatially local clusters of quadruples – or objects according to definition 2.1. A single quadruple can be used to estimate all parameters of an object's symmetry plane, $\boldsymbol{\pi} = (\boldsymbol{n}^\top d)^\top$. $\boldsymbol{n}$, the direction of symmetry, is calculated as per equation 7. A point on the symmetry plane is required to estimate d. In this case $\boldsymbol{P}$ and $\boldsymbol{Q}$ are estimated using triangulation, and d is obtained as:

$$d = -\frac{1}{2}(\boldsymbol{P} + \boldsymbol{Q})^\top \boldsymbol{n} \tag{8}$$

3.3 Finding Inliers for a Hypothesis

It is possible to find all quadruple "inliers" for a given symmetry plane hypothesis, $\boldsymbol{\pi} = (\boldsymbol{n}^\top d)^\top$, by examining the reprojection error of the four points in each quadruple as follows. First reconstruct 3D points $\boldsymbol{P}$ and $\boldsymbol{Q}$ from $\boldsymbol{p_1}$ and $\boldsymbol{q_1}$ by using the symmetry prior (equation 5). These 3D points are then reprojected into both image planes, as per equations 9 and 10, where δ is the distance between the two camera centers, and $\hat{\boldsymbol{p}}_{\boldsymbol{1}}$ is the reprojection of $\boldsymbol{P}$ in the first image plane and $\hat{\boldsymbol{p}}_{\boldsymbol{2}}$ in the second image plane.

$$\hat{\boldsymbol{p}}_{\boldsymbol{1}} = \frac{f}{P_z}\left(P_x, P_y\right)^\top \tag{9}$$

$$\hat{\boldsymbol{p}}_{\boldsymbol{2}} = \frac{f}{P_z}\left(P_x + \delta, P_y\right)^\top \tag{10}$$

The reprojection error of both reconstructed 3D points is then given by equation 11.

$$\text{reprojection\_error}_{\boldsymbol{P}} = ||\hat{\boldsymbol{p}}_{\mathbf{1}} - \boldsymbol{p}_{\mathbf{1}}|| + ||\hat{\boldsymbol{p}}_{\mathbf{2}} - \boldsymbol{p}_{\mathbf{2}}|| \tag{11}$$

If the reprojection error of both $\boldsymbol{P}$ and $\boldsymbol{Q}$ are below a specified threshold then the quadruple is considered an inlier to the object hypothesis.

An additional parameter, the "maximum object size" is used to alleviate noise from spatially distant quadruples that happen to be inliers to the object hypothesis. Recall that a point on the symmetry plane is calculated to get the d parameter in equation 8. A quadruple is considered an inlier only if both $\boldsymbol{P}$ and $\boldsymbol{Q}$ are within a specified distance to this initial point on the symmetry plane. This specified distance sets the expected maximum size of an object.

Note that if the symmetric reconstruction (equation 5) is performed on $\boldsymbol{p}_{\mathbf{1}}$ and $\boldsymbol{q}_{\mathbf{1}}$, then $\hat{\boldsymbol{p}}_{\mathbf{1}} = \boldsymbol{p}_{\mathbf{1}}$, and $\hat{\boldsymbol{q}}_{\mathbf{1}} = \boldsymbol{q}_{\mathbf{1}}$, and the reprojection error for these points is zero. However, points $\boldsymbol{P}$ and $\boldsymbol{Q}$ will be different than those computed from binocular disparities via triangulation. In particular, the distances between $\boldsymbol{P}$, $\boldsymbol{Q}$, and the camera center, are noisy in triangulation because of a combination of pixelation and the large ratio of the reconstructed depth to the distance between the camera centers. Equation 5 does not have this defect, and corrects the binocular reconstruction. Put differently, 3D symmetry allows for subpixel binocular reconstruction.

3.4 Non-linear Optimization of Hypotheses

Each symmetry plane has four parameters, $\boldsymbol{\pi} = (\boldsymbol{n}^\top d)^\top$, but only two degrees of freedom. The direction of symmetry, $\boldsymbol{n}$ is a unit normal, but it only has one degree of freedom because it must be orthogonal to the floor normal, as specified in equation 7. Nelder-Mead[38] is then used over this two-dimensional subspace to find the symmetry plane parameters that maximize the number of quadruple inliers.

3.5 Choosing Non-overlapping Hypotheses

The steps outlined above generate a single object hypothesis. In a procedure similar to RANSAC, we can create an arbitrary number of hypotheses. (The precise number is given in section 3.6.) Many of these hypotheses overlap spatially and must be discarded; however, choosing a maximal non-overlapping subset of object hypotheses is NP-hard. *Branch & bound*[39] was used to accomplish this step.

Let S be the set of initial object hypotheses generated according to the steps outlined above. Let $|s|$ be the number of quadruples in a given object hypothesis, $s \in S$. Then branch & bound is used to find the optimal set of hypotheses, $S^{'} \subseteq S$, as described by equation 12.

$$S^{'} = \underset{S^{'} \subseteq S}{argmax} \sum_{s \in S^{'}} |s| \quad \text{where } \texttt{intersect}(s_i, s_j) = 0 \quad \forall s_i, s_j \in S^{'}, i \neq j. \tag{12}$$

An important detail is calculating the spatial intersection between two object hypotheses: $\texttt{intersect}(s_i, s_j)$. Using the normal to the floor plane, we calculated the orthographic projection of each hypothesis onto the floor, and then found the 2D convex hull for the projected points. This convex hull is a 2D polygon representing the image of the object hypothesis on the floor. Two object hypotheses were considered overlapping if their 2D hulls overlapped.

3.6 Algorithm Parameters

The following parameters are used in the clustering algorithm.

Gaussian smoothing	$\sigma = 2.0$
ROI in search for quadruples	150 pixels
Colinearity test threshold	1.5 pixels
Quadruple inlier test	1.5 pixels
Maximum size of object	1.0 meter
Number of object hypotheses	80

These parameters were chosen based on features of the algorithm. Object size is the exception, and was set to be about 50% larger than the expected size of the largest object in clustering.

4 Experimental Results

The novelty of our approach makes it difficult to compare our algorithm to existing techniques. Unsupervised object localization is a form of unsupervised clustering, which our algorithm performs using symmetry as an informative prior. Therefore we considered it appropriate to compare our results to K-means because of its long history, and the typical usage of distance metrics for most data clustering techniques. Thus, we tested mirror symmetry clustering – as an informative prior for object localization – against K-means, a benchmark method based solely on spatial clustering.

A corpus was acquired to perform a comparison. 180 pairs of 1024x768 grayscale images were captured under normal indoor lighting conditions using a Point Grey Bumblebee2® stereo camera with a 12cm baseline, and a 66° horizontal field of view. Five to ten objects were featured in each scene (a room), where the objects were mostly polyhedral objects: toys, childrens furniture, a pram, a vacuum cleaner, and a tripod. 75 images featured a person. The floor was covered by uniformly colored carpet.

As described in section 3.1, our symmetry based approach to object localization relies on estimating the plane of the floor. For this estimate to be reliable, there must be sufficient samples of floor patches available in the images. Since the points on the floor are removed from further analysis, the type of texture on the floor is unimportant. Furthermore, our approach relies on binocular disparity information in order to solve the symmetry correspondence problem, as discussed in section 2.4. When the objects are too far from the camera, then the stereo system degenerates to single view geometry. For example, an object 0.3 meters deep, placed 4 meters from our stereo camera setup, has only 2 pixels of disparity difference between front and back. Therefore, when constructing the corpus, objects were placed between 1.5 and 4.5 meters from the stereo camera.

2D ground truth was specified as a set of bounding rectangles in the left-camera-image of each pair of images. Each bounding rectangle was drawn by hand around the regions containing the individual exemplars. The rectangles were axis-aligned such that they have horizontal and vertical edges.

4.1 K-Means

Since mirror symmetry is calculated in 3D, we applied K-means clustering to an unstructured 3D point-cloud. A disparity map was calculated as per section 3 of this paper. The symmetry based algorithm only considered binocular correspondences that coincided with the canny edge maps generated from each image. This sparse point cloud was used to reduce the search space for symmetrical correspondences; however, it also reduces noise from texture-based artifacts typical in stereo reconstructions. In order to do a fair comparison we restricted the “K-means” point cloud to the same set of 3D points that were used to generate two-view mirror symmetry quadruples. The method described by Caliński & Harabasz[40] was used to automatically determine the value of K:

the number of objects in the image. Once clusters were determined, the 3D points were projected back to the image plane of the left camera, and bounding rectangles with horizontal and vertical sides were calculated.

4.2 Comparison Function

Bounding rectangles were compared as follows. Intersection-over-union[41] (equation 13) was used to calculate the best matches between all rectangles representing an algorithm's output and all ground truth rectangles. The best matching pair of rectangles were paired together first, and then removed from analysis. This procedure was repeated recursively until all ground truth rectangles were matched.

$$\texttt{intersection\_over\_union}(A, B) = \frac{\texttt{area}(A \cap B)}{\texttt{area}(A \cup B)} \tag{13}$$

We averaged the scores for each image. Our method for scoring each image does not produce a penalty for estimating too many objects; however, if the algorithm estimated that there were too few objects, then some ground truth rectangles were scored as zero. As will be seen below, this fact favored, on average, K-means clustering. It follows that the observed superiority of our method in this analysis is a conservative estimate.

We also calculated F_1 statistics using the following labeling procedure. All rectangles were labeled as either *true positives* (TP) or *false positives* (FP), where a true positive was recorded if intersection-over-union with a ground-truth rectangle was greater than 0.5. If a ground truth rectangle was not paired with a true positive object hypothesis, then it was labeled as a *false* *negative* (FN). F_1 is then calculated according to equation 14.

$$F_1 = \frac{2\mathtt{TP}}{2\mathtt{TP} + \mathtt{FN} + \mathtt{FP}} \tag{14}$$

4.3 K-Means versus Symmetry Based Object Localization

Bounding rectangles for 3 representative examples are shown in Figure 3. In Figure 4 we show a histogram of the ratios of mean scores, as defined by equation 13, for our algorithm and for K-means across all 180 scenes. Ratios greater than 1 imply that our algorithm performed better.

Most ratios are greater than 1 indicating that our algorithm did perform better. We want to point out, however, that K-means also performed reasonably well. This good performance is partly the result of eliminating spurious 3D points in the front-end of our method, where we reconstruct only those 3D points that represent binocular correspondences for both texture patches and edges (see section 4.1). However, since the K-means method relies solely on a distance metric, it seems to produce localizations that bleed over multiple scene objects when the objects are close together in 3D. The symmetry based method is much more robust in this circumstance. Both methods work well if the objects are far apart from each other in 3D, even if one occludes the other in the camera images.

The symmetry based clustering also performed better than Caliński & Harabasz's method for determining the number of clusters. A comparison of the number of clusters detected between the two methods is given in Figure 5. In this figure we see that the Caliński & Harabasz heuristic tends to overestimate the numbers of objects (clusters) in the scenes. In contrast, symmetry based clustering tends to be more accurate, and to slightly underestimate the numbers of objects in the scenes.

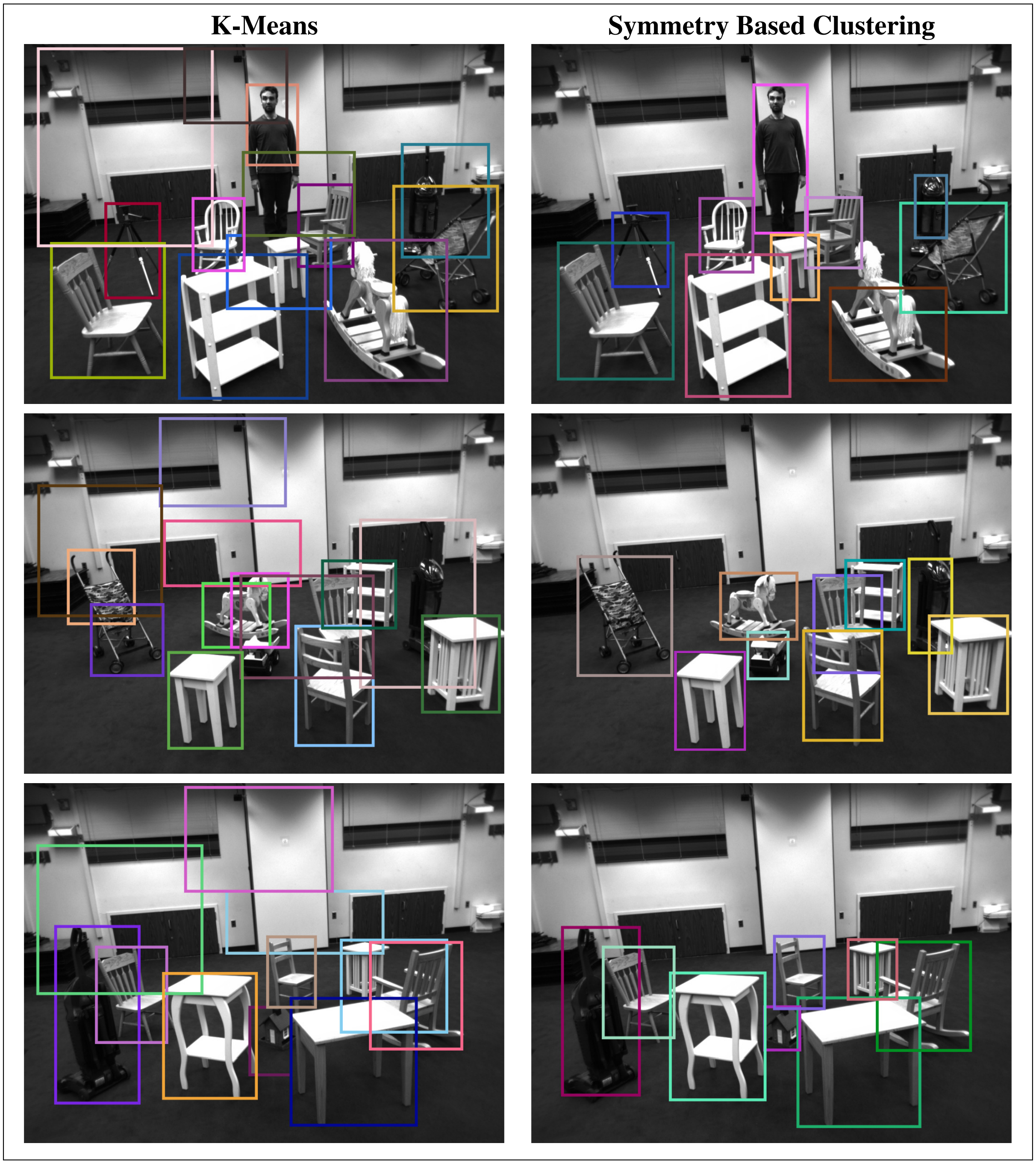

Fig 3: Representative results comparing K-means (left column) to symmetry based object localization (right column).

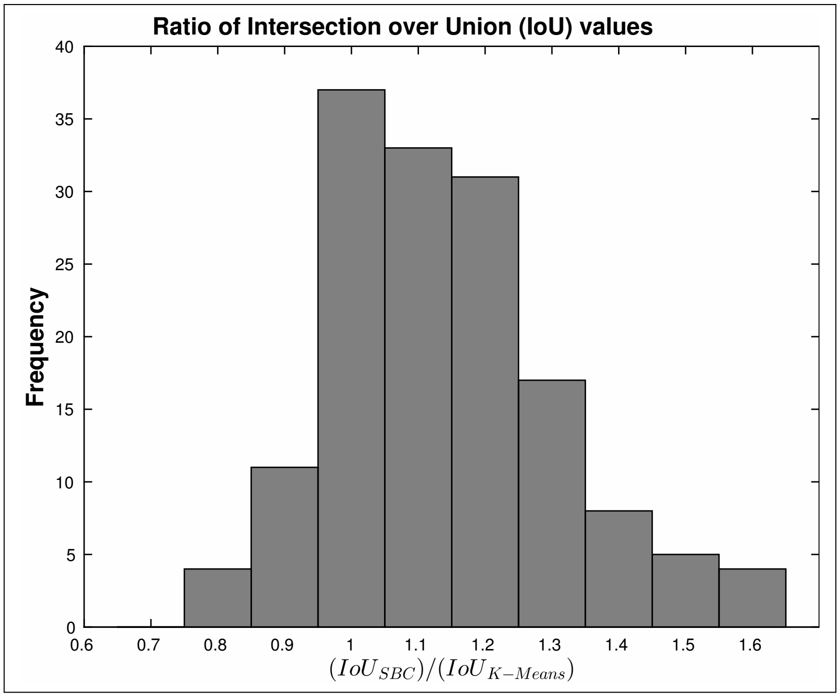

Fig 4: Histogram of the ratio of mean intersection-over-union of bounding rectangles for each image. A ratio of 1.0 implies that *symmetry based clustering* (SBC) and *K-means* performed equally well. Greater than 1.0 implies that SBC performed better.

Note, as described in section 4, the corpus was designed with mostly polyhedral objects placed close enough to the stereo camera such that symmetry correspondence could be solved from disparity information. Furthermore, as described in section 3.1, we constrained the symmetry plane to be orthogonal to the estimated floor plane – and the floor was clearly visible in every corpus image. The results reflect these controlled experimental conditions. In the conclusions section we discuss how to relax these constraints in order to develop a general purpose approach to symmetry based object localization.

4.4 Runtime Performance of Symmetry Based Object Localization

Runtime performance for the symmetry based technique was dominated by the final branch & bound step. After generating multiple overlapping object hypotheses, branch & bound was used to find a maximal set of non-overlapping objects, as described in section 3.5. The speed of this step

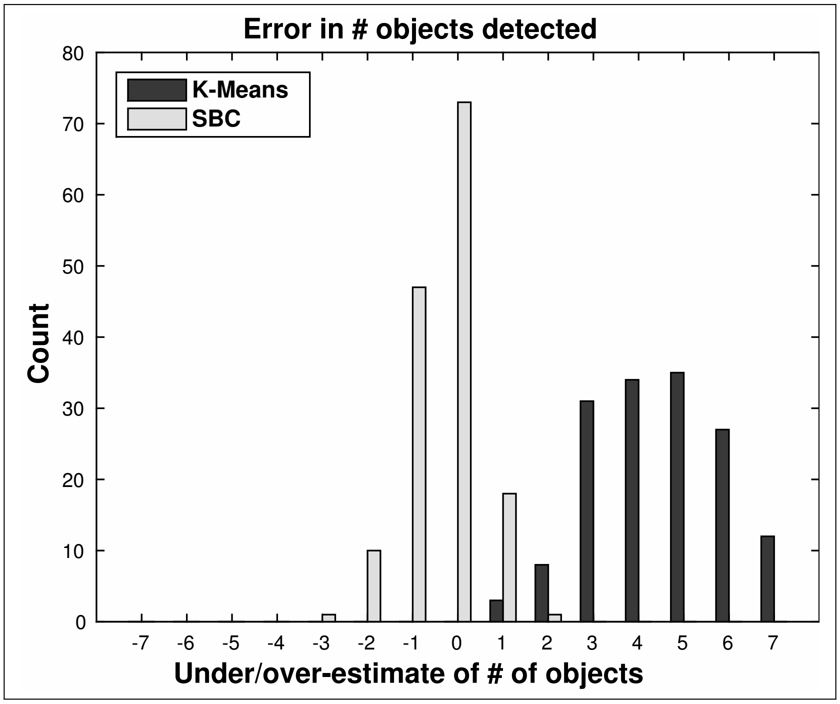

Fig 5: The difference between the number of clusters detected by *symmetry based clustering* (SBC), or by Caliński & Harabasz's method for determining K in K-means, and the actual number of objects. Zero error implies that a method detected the correct number of clusters (objects). Positive error means that a method detected more clusters than there were objects in the scene. Caliński & Harabasz shows a bias toward reporting too many objects under the experimental conditions. SBC is both more accurate, and slightly conservative, in its estimates of numbers of objects.

is determined by the ability of branch & bound to quickly find a good upper bound to equation 12, thereby allowing it to splice away exponentially sized chunks of the search space. As such, it is crucial to test the most promising hypotheses first, where object hypotheses are ordered according to the number of quadruples they contained. This approach was usually good enough to find an optimal set of non-overlapping objects without recourse to some form of non-maximal suppression.

In our initial experiments, however, we found that the branch & bound step would occasionally run for hours on cluttered input images. Thus we experimented with a simple mechanism that returned the best (sub-optimal) result after 30 minutes, about double the median time for the exhaustive search, in order to assess its effect on accuracy. Table 1 shows the F_1 score for the testing corpus, with and without timeout. We see that when the timeout was not used, both runtime and performance (as measured by F_1 score) increase as more hypotheses are generated. There is little effect of timeout, except for when the number of input hypotheses is large. Under the given experimental conditions, 80 hypotheses with 30 minute timeout gives near best performance for this method ($F_1 = 0.83$), with median runtime 6.55 minutes, and runtime bounded to 30 minutes.

There is another way to speed-up the processing time. In our experiment the scene contained images of multiple objects, each object occupying only a fraction of the camera image. It follows that the entire image could be divided into several smaller regions and our algorithm could then be applied to individual regions, one by one. Considering the NP-hard nature of our algorithm, the sum of processing times of smaller problems (regions) is likely to be less than the processing time of solving the entire problem (analyzing the entire image). What we are describing is the essence of divide and concur approach that has been used in many applications in the past. In order to evaluate how well this approach would work in our application, we looked at our results separately for the scenes containing 5 objects, 6 objects and so on. For 5 objects in the scene, 40 hypotheses led to precision $F_1 = 0.87$ with the median runtime 0.6 minutes and max runtime 1.3 min (see Table 2). Compare this to 10 objects in the scene. Ten objects required 320 hypotheses in order to produce $F_1 = 0.88$, but the resulting median time was 31 minutes and max runtime almost 12 hours. Clearly, detecting 5 objects at a time in a scene containing 10 objects would lead to substantially smaller runtime, compared to the case of detecting all 10 objects at once. This observation is not surprising, but it does suggest that our algorithm could use what is referred to in human vision literature as visual attention.[42] Directing visual attention towards particular regions in the image requires additional computational tools, such as saliency measures. There are a number of saliency measures in the literature, as well as models of visual search using eye movements. Our future work will examine these aspects of vision with the purpose of bringing

# Hypotheses	Timeout	Optimal	Runtime (min, median, max) minutes		
10	0.33	0.30	0.09	0.29	0.83
20	0.58	0.58	0.16	0.47	1.46
40	0.75	0.76	0.40	1.08	2.96
80	0.83	0.83	1.89	6.55	1986.35
160	0.87	0.89	3.74	11.12	613.52
320	0.60	0.90	4.76	16.29	2111.52

Table 1: F_1 with and without timeout. *Timeout* means that the algorithm returned the best result after 30 minutes of computation. *Optimal* means that the branch & bound step ran to completion, giving the optimal result. The given runtimes are for the optimal condition. The timeout did affect performance when the number of input hypotheses was large.

Table 2.A F_1 and runtime for 40 input hypotheses.

# Objects	F_1	Runtime (min, median, max) minutes		
5	0.87	0.40	0.61	1.34
6	0.81	0.46	0.83	1.83
7	0.82	0.55	1.09	2.16
8	0.70	0.91	1.26	2.58
9	0.74	0.86	1.42	2.87
10	0.65	0.83	1.40	2.96

Table 2.B F_1 and runtime for 320 input hypotheses.

# Objects	F_1	Runtime (min, median, max) minutes		
5	0.90	4.76	7.40	18.52
6	0.91	5.06	10.80	22.25
7	0.91	8.01	14.94	1117.42
8	0.93	10.86	16.63	308.10
9	0.89	13.89	25.24	2111.52
10	0.88	17.14	31.04	713.11

Table 2: F_1 and runtime as a function of the number of objects in a scene, for 40 (Table 2.A), and 320 (Table 2.B) input hypotheses, and with branch & bound running until completion. We see that runtime is much faster for fewer objects, suggesting that overall performance can be improved by adopting a divide and conquer approach.

our model of figure-ground organization closer to real time performance – the kind of performance that characterizes human vision.

5 Conclusions and Future Work

Our results suggest that the biologically motivated symmetry prior is useful in FGO (object localization). In a sense these results are not surprising because our new method is based on a straightforward rational argument, namely, that 3D symmetries uniquely identify 3D objects. Intuitively this argument makes a lot of sense, and insofar as there is a method for detecting 3D symmetries in a scene then the resulting FGO should be reliable. Our results support this claim; however, we believe that progress can be made by improving the front end of our model, where 3D symmetry is detected. A few suggestions for future research are listed below.

Firstly, this experiment uses a rudimentary definition of shape: definition 2.1. If an object's symmetries provide the informed prior that makes accurate FGO possible, then it stands to reason that a richer definition of an object's symmetries would produce even better results. In particular, individual object hypotheses would become more constrained, and more likely to appeal to the human intuition for shape. This, in turn, would prune the search space that the branch & bound step must traverse to find the optimal set of non-overlapping objects, thus improving the speed of the algorithm as well. Any heuristical method for finding non-overlapping objects should also benefit from a richer definition of shape as well.

Secondly, humans are able to solve the 3D symmetry correspondence problem from single 2D images, but how this occurs is currently a topic of active research.[43] Solving for object localization from single images would widen the practical applications of this approach. This is especially so for uncalibrated cameras, since a 3D aware approach to object localization could be used in general image databases, presenting a significant advance over the state of the art.

Thirdly, the procedure presented was designed and tested on mostly polyhedral objects, but this approach should be extended to smoothly curved surfaces. Polyhedral objects tend to have well defined edges in the binary edge maps produced by the canny operator. Importantly, these edges are sufficient for recovering the symmetry of the object. This simplifies the symmetry correspondence problem to searching for pairs of points on a binary edge map; however, such an approach may not work well with smooth surfaces. Our results show that "round" structures, such as people, can be localized using symmetry applied to binary edgemaps; however, this is a preliminary result, and it is unknown, for example, how well individual people can be localized in a crowd.

Note that once FGO is solved, the individual objects can be used as a saliency map to direct the "attention" of further processing steps. The topic of saliency maps was started 30 years ago[44] and remains an active field today. Symmetry based FGO should be useful to researchers who are interested in finding relationships between objects, and building event models for scenes, which is sometimes considered part of the vision problem.[42] In turn, higher level reasoning about a scene could disambiguate a symmetry based FGO approach. For example, a symmetrical collection of tables and chairs in a classroom could be considered as one object or many, depending on how the information needs to be used by an event model for the scene. Thus we are proposing that attention and saliency could work on two levels: as the front end to divide the image into subregions for solving FGO and later, after FGO is solved, to focus subsequent analyses on particular objects and groups of objects. This double status of attention may correspond to the "what" and "where" pathways in the human visual system.[45]

We believe that detecting and using 3D symmetry is an essential step in visual processing, because psychophysical experiments have already shown that symmetry is a powerful prior used by the human vision system in a variety of unsolved vision applications, such as figure-ground organization, 3D shape recovery, and shape constancy.[6] Although the present work is preliminary, our experiment suggests an avenue towards bridging the divide between human and computer performance on object localization.

Acknowledgments

This research was supported by the National Eye Institute of the National Institutes of Health under award number 1R01EY024666-01. The content of this article is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. We would like to thank the reviewers for their helpful feedback, and Vikrant Satheesh for annotating the testing corpus.

References

1 K. Koffka, *Principles of Gestalt psychology*, Routledge (2013).

2 E. Rubin, "Synsoplevede figurer (visually experienced figures)," *Copenhagen: Gyldendal* (1915).

3 J. Wagemans, J. Elder, M. Kubovy, S. Palmer, M. Peterson, M. Singh, and R. von der Heydt, "A century of gestalt psychology in visual perception: I. perceptual grouping and figure–ground organization.," *Psychological Bulletin* **138**(6), 1172 (2012).

4 J. Wagemans, J. Feldman, S. Gepshtein, R. Kimchi, J. Pomerantz, P. van der Helm, and C. van Leeuwen, "A century of gestalt psychology in visual perception: Ii. conceptual and theoretical foundations.," *Psychological Bulletin* **138**(6), 1218 (2012).

5 T. Poggio, V. Torre, and C. Koch, "Computational vision and regularization theory," *Nature* **317**(6035), 314–319 (1985).

6 Z. Pizlo, Y. Li, T. Sawada, R. Steinman, *et al.*, *Making a machine that sees like us*, Oxford University Press (2014).

7 Y. Li, T. Sawada, Y. Shi, T. Kwon, and Z. Pizlo, "A bayesian model of binocular perception of 3d mirror symmetrical polyhedra," *Journal of Vision* **11**(4), 11 (2011).

8 Y. Li, T. Sawada, Y. Shi, R. Steinman, and Z. Pizlo, "Symmetry is the sine qua non of shape," in *Shape perception in human and computer vision*, 21–40, Springer (2013).

9 S. Levinson, "Frames of reference and molyneuxs question: Crosslinguistic evidence," *Language and Space* , 109–169 (1996).

10 L. Talmy, *Toward a cognitive semantics*, MIT press (2003).

11 E. Pecoits, K. Konhauser, N. Aubet, L. Heaman, G. Veroslavsky, R. Stern, and M. Gingras, “Bilaterian burrows and grazing behavior at 585 million years ago,” *Science* **336**(6089), 1693–1696 (2012).

12 T. Sawada, Y. Li, and Z. Pizlo, “Shape perception,” *The Oxford Handbook of Computational and Mathematical Psychology* , 255 (2015).

13 D. Marr and H. Nishihara, “Representation and recognition of the spatial organization of three-dimensional shapes,” *Proceedings of the Royal Society of London B: Biological Sciences* **200**(1140), 269–294 (1978).

14 T. Binford, “Visual perception by computer,” in *IEEE conference on Systems and Control*, **261**, 262 (1971).

15 S. Carlsson, *Computer Vision — ECCV’98: 5th European Conference on Computer Vision*, 249–263. Springer Berlin Heidelberg, Berlin, Heidelberg (1998).

16 L. van Gool, T. Moons, and M. Proesmans, “Mirror and point symmetry under perspective skewing,” in *1996 Conference on Computer Vision and Pattern Recognition (CVPR ’96), June 18-20, 1996 San Francisco, CA, USA*, 285–292 (1996).

17 W. Hong, A. Yang, K. Huang, and Y. Ma, “On symmetry and multiple-view geometry: Structure, pose, and calibration from a single image,” *International Journal of Computer Vision* **60**(3), 241–265 (2004).

18 G. Gordon, “Shape from symmetry,” in *1989 Advances in Intelligent Robotics Systems Conference*, 297–308, International Society for Optics and Photonics (1990).

19 P. Cicconi and M. Kunt, “Symmetry-based image segmentation,” in *Berlin-DL tentative*, 378–384, International Society for Optics and Photonics (1993).

20 D. Sharvit, J. Chan, H. Tek, and B. Kimia, "Symmetry-based indexing of image databases," *Journal of Visual Communication and Image Representation* **9**(4), 366–380 (1998).

21 S. Utcke and A. Zisserman, "Projective reconstruction of surfaces of revolution," in *Pattern Recognition*, 265–272, Springer (2003).

22 C. Colombo, A. Bimbo, and F. Pernici, "Metric 3d reconstruction and texture acquisition of surfaces of revolution from a single uncalibrated view," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27**(1), 99–114 (2005).

23 C. Phillips, M. Lecce, C. Davis, and K. Daniilidis, "Grasping surfaces of revolution: Simultaneous pose and shape recovery from two views," in *IEEE International Conference on Robotics and Automation, ICRA*, 1352–1359 (2015).

24 T. Cham and R. Cipolla, "Symmetry detection through local skewed symmetries," *Image and Vision Computing* **13**(5), 439–450 (1995).

25 S. Sinha, K. Ramnath, and R. Szeliski, "Detecting and reconstructing 3d mirror symmetric objects," in *Computer Vision - ECCV'12: 12th European Conference on Computer Vision*, 586–600 (2012).

26 K. Köser, C. Zach, and M. Pollefeys, "Dense 3d reconstruction of symmetric scenes from a single image," in *Pattern Recognition*, 266–275, Springer (2011).

27 S. Dickinson, "Challenge of image abstraction," *Object categorization: computer and human vision perspectives* , 1 (2009).

28 J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman, "Discovering object categories in image collections," (2005).

29 K. Grauman and T. Darrell, "Unsupervised learning of categories from sets of partially matching image features," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006)*, 19–25 (2006).

30 T. Deselaers, B. Alexe, and V. Ferrari, "Localizing objects while learning their appearance," in *Computer Vision - ECCV'10 11th European Conference on Computer Vision*, 452–466, Springer (2010).

31 A. Faktor and M. Irani, ""clustering by composition"–unsupervised discovery of image categories," in *Computer Vision - ECCV'12 12th European Conference on Computer Vision*, 474–487, Springer (2012).

32 M. Cho, S. Kwak, C. Schmid, and J. Ponce, "Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals," *arXiv preprint arXiv:1501.06170* (2015).

33 G. Kim and A. Torralba, "Unsupervised detection of regions of interest using iterative link analysis," in *Advances in Neural Information Processing Systems*, 961–969 (2009).

34 A. Jain, "Data clustering: 50 years beyond k-means," *Pattern Recognition Letters* **31**(8), 651–666 (2010).

35 C. Rothwell, D. A. Forsyth, A. Zisserman, and J. L. Mundy, "Extracting projective structure from single perspective views of 3d point sets," in *Fourth International Conference on Computer Vision, ICCV 1993*, 573–582 (1993).

36 R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, Cambridge University Press (2003).

37 Y. Li, T. Sawada, L. Latecki, R. Steinman, and Z. Pizlo, "A tutorial explaining a machine

vision model that emulates human performance when it recovers natural 3d scenes from 2d images," *Journal of Mathematical Psychology* **56**(4), 217–231 (2012).

38 D. Olsson and L. Nelson, "The nelder-mead simplex procedure for function minimization," *Technometrics* **17**(1), 45–51 (1975).

39 A. Land and A. Doig, "An automatic method of solving discrete programming problems," *Econometrica: Journal of the Econometric Society* , 497–520 (1960).

40 T. Caliński and J. Harabasz, "A dendrite method for cluster analysis," *Communications in Statistics-Theory and Methods* **3**(1), 1–27 (1974).

41 M. Everingham., L. van Gool, C. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision* **88**, 303–338 (2010).

42 J. K. Tsotsos, *A computational perspective on visual attention*, MIT Press (2011).

43 V. Jayadevan, "In preparation,"

44 C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry," in *Matters of Intelligence*, 115–141, Springer (1987).

45 M. Mishkin, L. Ungerleider, and K. Macko, "Object vision and spatial vision: two cortical pathways," *Trends in Neurosciences* **6**, 414–417 (1983).

Aaron Michaux is a PhD candidate in the department of Electrical and Computer Engineering at Purdue University. He has a BS in computer science (2001) from the University of Queensland, and a BA in psychology (2010) from Saint Thomas University. He is the author on six journal papers. In his research he takes a cognitive science approach to engineering questions in computer vision. He is a member of IEEE.

Vijai Jayadevan Vijai Jayadevan obtained his bachelors degree from Cochin University in 2008 and masters degree from The University of Arizona in 2013, both in electrical engineering. He is currently working towards his PhD degree in electrical engineering at Purdue University. His research interests include computer vision, computer graphics, machine learning and signal processing. He is a student member of IEEE.

Zygmunt Pizlo is a Professor of Psychological Sciences at Purdue University. He received his Ph.D. degree in Electronics in 1982 and his Ph.D. degree in Psychology in 1991. His research focuses on 3D vision with special emphasis on 3D shape. He published two monographs on these subjects in 2008 and 2014. His broader interests include computational modeling of cognitive functions such as problem solving and motor control as well as image and video processing applications.

Edward J. Delp is currently The Charles William Harrison Distinguished Professor of Electrical and Computer Engineering at Purdue University. His research interests include image analysis, computer vision, image and video compression, multimedia security, medical imaging, communication and information theory. Dr. Delp is a Fellow of the IEEE, a Fellow of the SPIE, a Fellow of the Society for Imaging Science and Technology (IS&T), and a Fellow of the American Institute of Medical and Biological Engineering.