Chapter 12

Perceiving 3D from 2D Images

This chapter investigates phenomena that allow 2D image structure to be interpreted in terms of 3D scene structure. Humans have an uncanny abilty to perceive and analyze the structure of the 3D world from visual input. Humans operate effortlessly and often have little idea what the mechanisms of visual perception are. Before proceeding, three points must be emphasized. First, while the discussion appeals to analytical reasoning, humans readily perceive structure without conscious reasoning. Also, many aspects of human vision are still not clearly understood. Second, although we can nicely model several vision cues separately, interpretation of complex scenes surely involves competitive and cooperative processes using multiple cues simultaneously. Finally, our interest need not be in explaining human visual behavior at all, but instead be in solving a particular application problem in a limited domain, which allows us to work with simpler sets of cues.

The initial approach used in this chapter is primarily descriptive. The next section discusses the *intrinsic image*, which is an intermediate 2D representation that stores important local properties of the 3D scene. Then we explore properties of texture, motion, and shape that allow us to infer properties of the 3D scene from the 2D image. Although the emphasis of this chapter is more on identifying sources of information rather than mathematically modeling them, the final sections do treat mathematical models. Models are given for perspective imaging, computing depth from stereo, and for relating field of view to resolution and blur via the thin lens equation. Other mathematical modeling is left for Chapter 13.

12.1 Intrinsic Images

It is convenient to think of a 3D scene as composed of object surface elements that are illuminated by light sources and that project as regions in a 2D image. Boundaries between 3D surface elements or changes in the illumination of these surface elements result in contrast edges or *contours* in the 2D image. For simple scenes, such as those shown in Figures 12.1 and 12.2, all surface elements and their lighting can be represented in a description of the scene. Some scientists believe that the major purpose of the lower levels of the human visual system is to construct some such representation of the scene as the base for further processing. This is an interesting research question, but we do not need an answer in order to proceed with our work. Instead, we will use such a representation for scene and image

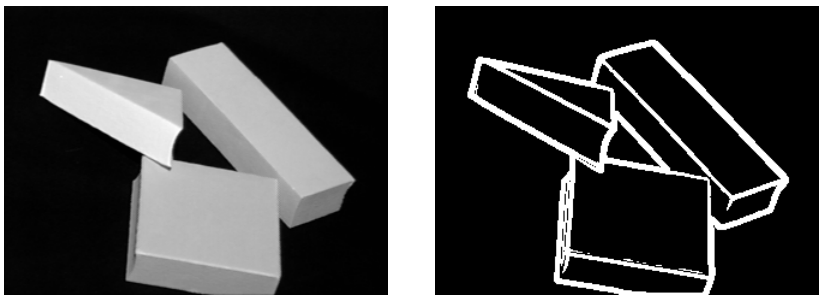


Figure 12.1: (Left) Intensity image of three blocks and (right) result of 5x5 Prewitt edge operator. Original image courtesy of Deborah Trytten.

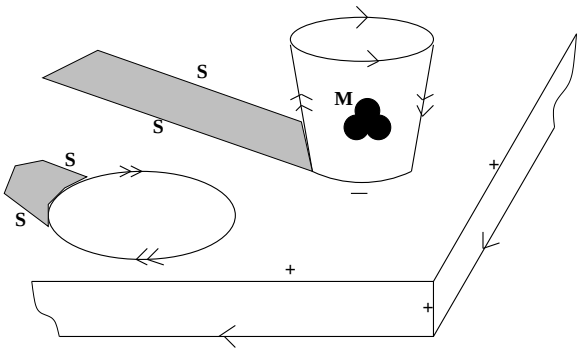


Figure 12.2: A 2D image with contour labels relating 2D contrasts to 3D phenomena such as surface orientation and lighting. Surface *creases* are indicated by + or -, an arrowhead (>) indicates a *blade* formed by the surface to its right while a double arrowhead indicates a smooth *limb* to its right, shadow boundaries are indicated by **S**, and reflectance boundaries are indicated by **M**.

description and machine analysis without regard to whether or not such a representation is actually computed by the human visual system.

Figure 12.2 shows an egg and an empty thin cup near the corner of a table. For this viewpoint, both the egg and cup occlude the planar table. Arrows along the region edges show which surface element occludes the other. The direction of the arrow is used to indicate which is the occluding surface; by convention, it is to the right as the edge is followed in the direction of the arrow. A single arrowhead (>) indicates a *blade*, such as the blade of a knife, where the orientation of the occluding surface element does not change much as the edge is approached; as the edge is crossed, the orientation of the occluded surface has no relation to that of the occluding surface. All of the object outlines in the right image of Figure 12.1 are due to blades. In Figure 12.2 a blade is formed at the lower table edge because the table edge, a narrow planar patch, occludes an unknown background. The top edge of the (thin) paper cup is represented as a blade because that surface occludes the background and has a consistent surface orientation as the boundary is approached. A more interesting case is the blade representing the top front cup surface occluding the cup interior.

A *limb* ($\gg$) is formed by viewing a smooth 3D object, such as the limb of the human body; when the edge of a limb boundary is approached in the 2D image, the orientation of the corresponding 3D surface element changes and approaches the perpendicular to the line of sight. The surface itself is *self-occluding*, meaning that its orientation continues to change smoothly as the 3D surface element is followed behind the object and out of the 2D view. A blade indicates a real edge in 3D whereas a limb does not. All of the boundary of the image of the egg is a limb boundary, while the cup has two separate limb boundaries. As artists know, the shading of an object darkens when approaching a limb away from the direction of lighting. Blades and limbs are often called *jump edges*: there is an indefinite jump in depth (range) from the occluding to occluded surface behind. Looking ahead to Figure 12.10 one can see a much more complex scene with many edge elements of the same type as in Figure 12.2. For example, the lightpost and light have limb edges and the rightmost edge of the building at the left is a blade.

Exercise 187

Put a cup on your desk in front of you and look at it with one eye closed. Use a pencil touching the cup to represent the normal to the surface and verify that the pencil is perpendicular to your line of sight.

Creases are formed by abrupt changes to a surface or the joining of two different surfaces. In Figure 12.2, creases are formed at the edge of the table and where the cup and table are joined. The surface at the edge of the table is convex, indicated by a '+' label, whereas the surface at the join between cup and table is concave, indicated by a '-' label. Note that a machine vision system analyzing bottom-up from sensor data would not know that the scene contained a cup and table; nor would we humans know whether or not the cup were glued to the table, or perhaps even have been cut from the same solid piece of wood, although our experience biases such top-down interpretations! Creases usually, but not always, cause a significant change of intensity or contrast in a 2D intensity image because one surface usually faces more directly toward the light than does the other.

Exercise 188

The triangular block viewed in Figure 12.1 results in six contour segments in the edge image. What are the labels for these six segments?

Exercise 189

Consider the image of the three machine parts from Chapter 1. (Most, but not all, of the image contours are highlighted in white.) Sketch all of the contours and label each of them. Do we have enough labels to interpret all of the contour segments? Are all of our available labels used?

Two other types of image contours are not caused by 3D surface shape. The *mark* ('M') is caused by a change in the surface albedo; for example, the logo on the cup in Figure 12.2 is a dark symbol on lighter cup material. Illumination boundaries ('I'), or shadows ('S'), are caused by a change in illumination reaching the surface, which may be due to some shadowing by other objects.

We summarize the surface structure that we're trying to represent with the following definitions. It is very important to understand that we are representing 3D scene structure as seen in a particular 2D view of it. These 3D structures *usually, but not always* produce detectable contours in case of a sensed intensity image.

90 DEFINITION *A* **crease** *is an abrupt change to a surface or a join between two different surfaces. While the surface points are continuous across the crease, the surface normal is discontinuous. A surface geometry of a crease may be observed from an entire neighborhood of viewpoints where it is visible.*

91 DEFINITION *A* **blade** *corresponds to the case where one continuous surface occludes another surface in its background: the normal to the surface is smooth and continues to face the view direction as the boundary of the surface is approached. The contour in the image is a smooth curve.*

92 DEFINITION *A* **limb** *corresponds to the case where one continuous surface occludes another surface in its background: the normal to the surface is smooth and becomes perpendicular to the view direction as the contour of the surface is approached, thus causing the surface to occlude itself as well. The image of the boundary is a smooth curve.*

93 DEFINITION *A* **mark** *is due to a change in reflectance of the surface material; for example, due to paint or the joining of different materials.*

94 DEFINITION *An* **illumination boundary** *is due to an abrupt change in the illumination of a surface, due to a change in lighting or shadowing by another object.*

95 DEFINITION *A* **jump edge** *is a limb or blade and is characterized by a depth discontinuity across the edge(contour) between an occluding object surface and the background surface that it occludes.*

Exercise 190 Line labeling of the image of a cube.

Draw a cube in general position so that the picture shows 3 faces, 9 line segments, and 7 corners. (a) Assuming that the cube is floating in the air, assign one of the labels from $\{+, -, >, or \gg\}$ to each of the 9 line segments, which gives the correct 3D interpretation for the phenomena creating it. (b) Repeat (a) with the assumption that the cube lies directly on a planar table. (c) Repeat (a) assuming that the cube is actually a thermostat attached to a wall.

Exercise 191 Labeling images of common objects.

Label the line segments shown in Figure 12.3: an unopened can of brand X soda and an open and empty box are lying on a table.

In Chapter 5 we studied methods of detecting contrast points in intensity images. Methods of tracking and representing contours were given in Chapter 10. Unfortunately, several 3D phenomena can cause the same kind of effect in the 2D image. For example, given a 2D contour tracked in an intensity image, how do we decide if it is caused by viewing an

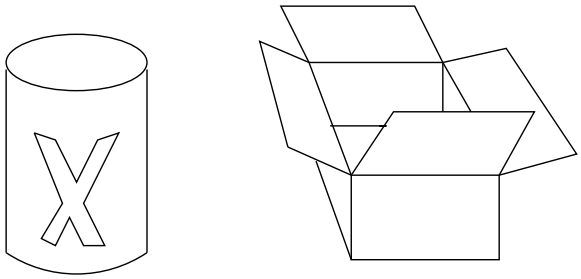


Figure 12.3: (left) an unopened can of Brand X Soda, which is a solid blue can with a single large orange block character 'X'; (right) an empty box with all four of its top flaps open, so one can see part of the box bottom that is not occluded by the box sides.

actual object or another object's shadow? Consider, for example, an image of a grove of trees taken on a sunny day. (Or, refer to the image of the camel on the beach toward the end of Chapter 5, where the legs of the camel provide the phenomena.) The shadows of the trees on the lawn ('S') may actually be better defined by our edge detector than are the limb boundaries ($\gg$) formed by the tree trunks. In interpreting the image, how do we tell the difference between the image of the shadow and the image of the tree; or, between the image of the shadow and the image of a sidewalk?

Exercise 192

Relating our work in Chapter 5 to our current topic, explain why the shadow of a tree trunk might be easier to detect in an image compared to the tree trunk itself.

Some researchers have proposed developing a sensing system that would produce an *intrinsic image*. An intrinsic image would contain four intrinsic scene values in each pixel.

- **range** or **depth** to the scene surface element imaged at this pixel
- **orientation** or **surface normal** of the scene element imaged at this pixel
- **illumination** received by the surface element imaged at this pixel
- **albedo** or surface reflectance of the surface element imaged at this pixel

Humans are good at making such interpretations for each pixel of an image given their surrounding context. Automatic construction of an intrinsic image is still a topic of research, but it is not being pursued as intensively as in the past. Many image analysis tasks do not need an intrinsic image. Chapter 13 will treat some methods useful for constucting intrinsic images or partial ones. An example of the intrinsic image corresponding to the image in Figure 12.2 is shown in Figure 12.4. The figure shows only the information from a small band of the intrinsic image across the end of the egg. The depth values show a gradual change across the table, except that at the edge of the table the change is more rapid, and there is a jump where the surface of the egg occludes the table. The orientation, or normal, of the table surface is the same at all the points of the table top; and, there is an abrupt change at the edge of the table. The orientation of the surface of the egg changes smoothly from one point to the next. The albedo values show that the table is a darker (5) material than the egg (9). The illumination values record the difference between table pixels that

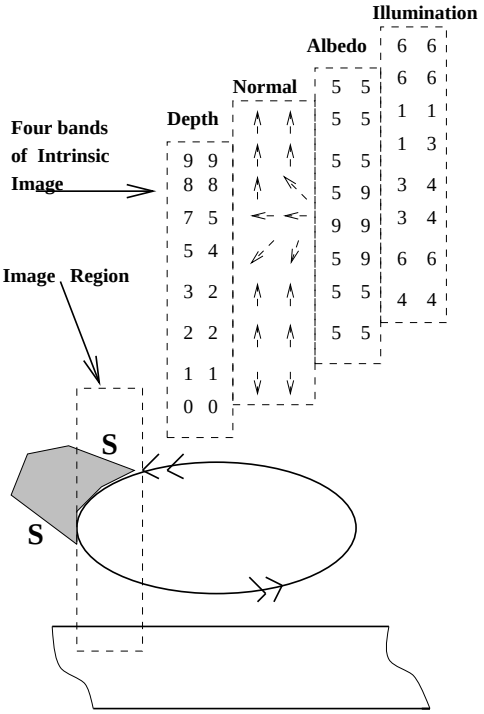


Figure 12.4: Intrinsic image corresponding to a small band across the egg of Figure 12.2. Each pixel contains four values representing surface depth, orientation, illumination, and albedo. See text for details.

are in shadow (1) versus those that are not. Similarly, pixels from the egg surface that is curving away from the illumination direction, assumed to be from the upper right, appear darker (3) than those directly facing the light because they receive less light energy per unit area.

Exercise 193 Line labeling an image of an outdoor scene.

Refer to the picture taken in Quebec City shown in Chapter 2. Sketch some of the major image contours visible in this image and label them using the label set $\{I/S, M, +, -, >, or \gg\}$

12.2 Labeling of Line Drawings from Blocks World

The structure of contours in an image is strongly related to the structure of 3D objects. In this section, we demonstrate this in a *microworld* containing restricted objects and viewing conditions. **We assume that the universe of 3D objects are those with trihedral corners: all surface elements are planar faces and all corners are formed by the intersection of exactly three faces.** The block in Figure 12.6 is one such object. We use the terms *faces*, *creases* and *corners* for the 3D structures and we use the terms *regions*, *edges*, and *junctions* for the images of those structures in 2D. A 2D image of the

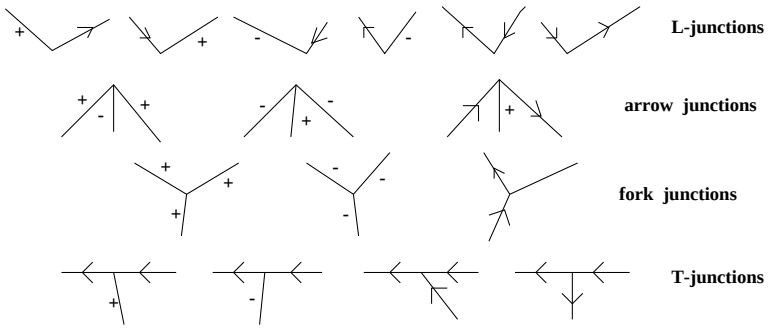


Figure 12.5: The only 16 topologically possible line junctions for images of trihedral blocks world (all 3D corners are formed by intersecting 3 planes and the object is viewed in general position). Junction types are, from top to bottom, **L**-junctions, arrows, forks, and **T**-junctions.

3D blocks world is assumed to be a line drawing consisting of regions, edges, and junctions. Moreover, we make the assumption that small changes in the viewpoint creating the 2D image cause no changes in the topology of this line drawing; that is, no new faces, edges, or junctions can appear or disappear. Often it is said that the object is "in general position".

Although our blocks microworld is so limited that it is unrealistic, the methods developed in this context have proven to be useful in many real domains. Thus we will add to the set of algorithms developed in Chapter 11 for matching and interpretation. Also, the blocks domain has historical significance and supports an intuitive development of the new methods.

From the previous section, we already know how to label the image edges using the labels $\{+, -, >\}$ to indicate which are creases or blades according to our interpretation of the 3D structure. No limbs are used since they do not exist in the blocks world. About 30 years ago, it was discovered that the possible combinations of line labels forming junctions is strongly constrained. In total, there are only 16 such combinations possible: these are shown in Figure 12.5. Figure 12.6 shows how these junction configurations appear in two distinct 3D interpretations of the same 2D line drawing.

There are four types of junctions according to the number of edges joining and their angles: for obvious reasons, they're called *L's, arrows, forks, and T's* from top to bottom by rows in Figure 12.5. Figure 12.6 shows an example with all four junction types. The junction marked **J** is an instance of the leftmost **L**-junction shown at the top of the catalogue in Figure 12.5, whereas the junction marked **C** is an instance of the second **L**-junction from the top right. **G** is the rightmost arrow-junction in the second row of Figure 12.5. There is only one **T**-junction, marked **D** in the figure. Note that, as Figure 12.5 shows, the occluding edge (cross) of the **T**-junction places no constraint on the occluded edge; all four possibilities remain, as they should. The four arrows in the block at the left (**B,E,G,I**) all have the same (convex) structure; however, the block at the right in Figure 12.6 has one other (concave) type of arrow-junction (**7**), indicating the convexity formed by the join of the block and wall.

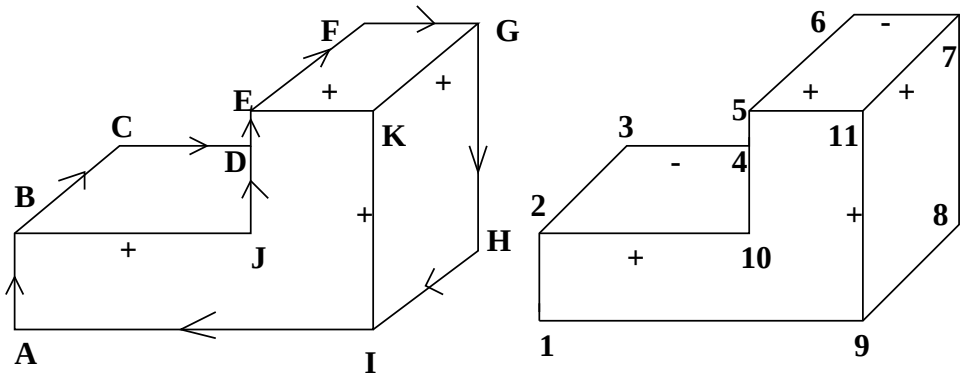


Figure 12.6: Two different interpretations for the same line drawing: (left) block floating in space and (right) block glued to back wall. The blade labels, omitted from the right figure, are the same as on the left.

Before proceeding, the reader should be convinced that all of the 16 junctions are, in fact, derivable from projections of 3D blocks. It is more difficult to reason that there are no other junctions possible: this has been proven, but for now, the reader should just verify that no others can be found while doing the exercises below.

Exercise 194

Label the lines in the right part of Figure 12.1 according to how your visual system interprets the scene at the left.

Exercise 195

Try to label all edges of all the blocks in Figure 12.7 as creases or blades. Every junction must be from the catalogue in Figure 12.5. (a) Which line drawings have consistent labelings? (b) Which drawings seem to correspond to a real object but cannot be labeled: why does the labeling fail? (c) Which drawings seem to correspond to impossible objects? Can any of these drawings be labeled consistently?

Two algorithmic approaches introduced in Chapter 11 can be used to automatically label such line drawings; one is sequential backtracking and the other is parallel relaxation labeling. We first formalize the problem to be solved: **given a 2D line drawing with a set of edges P_i (the observed objects), assign each edge a label L_j (the model objects) which interprets its 3D cause, and such that the combinations of labels formed at the junctions belong to the junction catalogue.** The symbols P and L have been used to be consistent with Chapter 11, which should be consulted for algorithm details. Coarse algorithm designs are given below to emphasize certain points. Both algorithms often produce many interpretations unless some external information is provided. A popular choice is to label all edges on the convex hull of the drawing as $>$ such that the hull is to the right.

If possible, the edges with the most constrained labels should be assigned first: it may even be that outside information (say stereo) already indicates that the edge corresponds

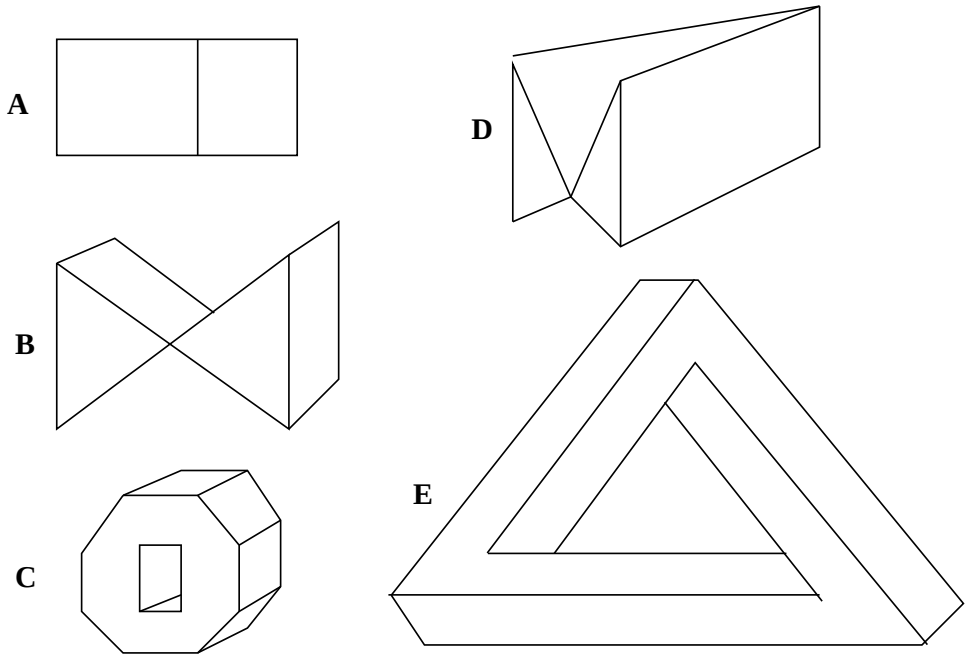


Figure 12.7: Line drawings which may or may not have 3D interpetations in our limited blocks world: which do and which do not and why?

Exercise 196

Sketch and label the line drawing of a real scene that yields at least two different instances of each of the four junction types. Create your own scene: you may use any structures from the several figures in this section.

Assign consistent interpretations to all edges of a scene graph.

Input: a graph representing edges E and junctions V.
Output: a mapping of edge set E onto label set $L = \{ +, -, >, < \}$.

- Assume some order on the set of edges, which may be arbitrary: $\mathbf{E} = \{P_1, P_2, ..., P_n\}$.
- At forward stage i, try labeling edge P_i using the next untried label from label set $\mathbf{L} = \{ +, -, >, < \}$.
- Check the consistency of the new label with all other edges that are adjacent to it via a junction in **V**.
- If a newly assigned label produces a junction that is not in the catalog, then backtrack; otherwise, try the next forward stage.

Algorithm 37: Labeling the edges of Blocks via Backtracking

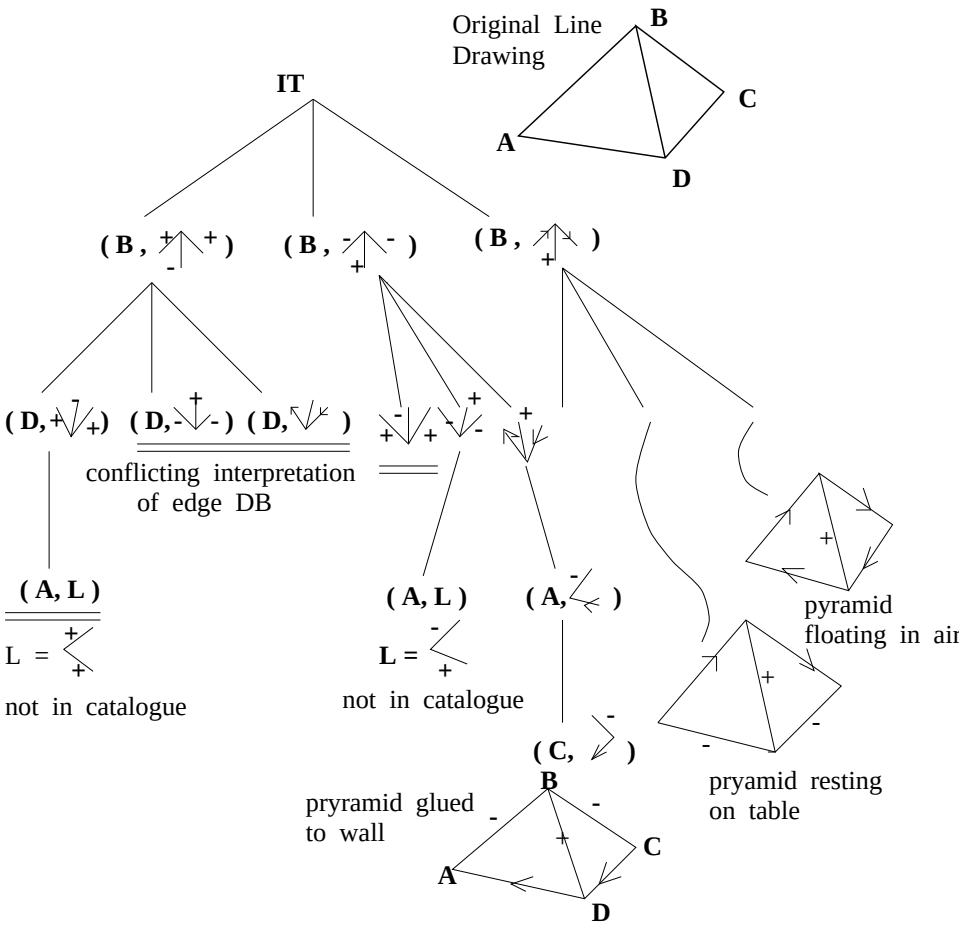


Figure 12.8: Interpretation Tree for the line drawing of a pyramid shown at the top right. At each level of the tree, one of the four junctions is labeled with one of the 16 junction labels from the catalogue in Figure 12.5. At the first level of the tree, an interpretation is assigned to junction **B**; subsequent levels assign interpretations to junctons **D**, **A**, and **C**. Three paths complete to yield the three interpretations shown at the bottom right.

to a 3D crease, for example. Some preprocessing should be done to determine the type of each junction according to the angles and number of incident edges. Other versions of this approach assign catalog interpretations to the junction labels and proceed to eliminate those that are inconsistent across an edge with the neighboring junction interpetations. Figure 12.8 shows an interpretation tree for interpreting the line drawing of a pyramid with four faces. The search space is rather small, demonstrating the strong constraints in trihedral blocks world.

Exercise 197

Complete the IT shown in Figure 12.8 by providing all the edges and nodes omitted from the right side of the tree.

Exercise 198

Construct the 5-level IT to assign consistent labels to all the edges of the pyramid shown in Figure 12.8. First, express the problem using the consistent labeling formalism from Chapter 11; define P, L, R_P, R_L, using the 5 observed edges and the 4 possible edge labels. Secondly, sketch the IT. Are there three completed paths corresponding to the three completed paths in Figure 12.8?

Line Labeling via Relaxation

As we have studied in Chapter 11, a discrete relaxation algorithm can also be used to constrain the interpretations of the parts of the line drawing. Here we assign labels to the edges of the line drawing: a similar procedure can be used to assign labels to the junctions.

Assign consistent interpretations to all edges of a scene graph.

Input: a graph representing edges E and junctions V.
Output: a mapping of edge set E onto *subsets of* label set $L = \{+, -, >, <\}$.

- Initally assign all labels $\{+, -, >, <\}$ to the label set of every edge P_i.
- At every stage, filter out possible edge labels by working on all edges as follows:
  - If a label L_j cannot form a legal junction using the possible labels for the edges connected to edge P_i, then eliminate label L_j from the label set of P_i.
- Stop iterating when no more label sets decrease in size.

Algorithm 38: Labeling the edges of Blocks via Discrete Relaxation

The above algorithm is a very simple representative of a large area of work with many variations. Because its simple operations can be done in any order, or even in parallel within each stage, the paradigm makes an interesting model for what might be happening in the human neural network downstream from the retina. Researchers have studied how to incorporate constraints from intensity and how to work at multiple resolutions. The blocks world work is very interesting and has been fruitful in its extensions. However, in the "toy form" presented here, it is useless for almost all real scenes, because (a) most 3D objects do not satisfy the assumptions and (b) actual 2D image representations are typically quite far from the line drawings required. Extensions, such as extending the label and junction catalogues to handle objects with curved surfaces and fixing errors in the line drawing, have been made and are mentioned in the references.

Exercise 199 Demonstrating the Necker Phenomena

This exercise is a variation on a previous one concerning the labeling of the edges of the image of a pyramid in general position (see previous exercise). Figure 12.9 shows a *wireframe* drawing of a cube: there is no occlusion and all 12 creases are visible as edges in the image. (a) Stare at the leftmost of the three drawings. Does your visual system produce a 3D interpretation of the line drawing? Does this interpretation change over a few minutes of staring? (b) Label the center drawing so that junction G is the image of a front corner. Cross out junction H and the three edges incident there so that a solid cube is represented. (c) Repeat (b) except this time have junction H be the image of a front corner and delete the edges incident on G. *Note that the 3D wireframe object is NOT a legal object in the blocks world we defined. However, we can use the same reasoning in interpreting small neighborhoods about any of the cube corners, which do belong to the junction catalog.*

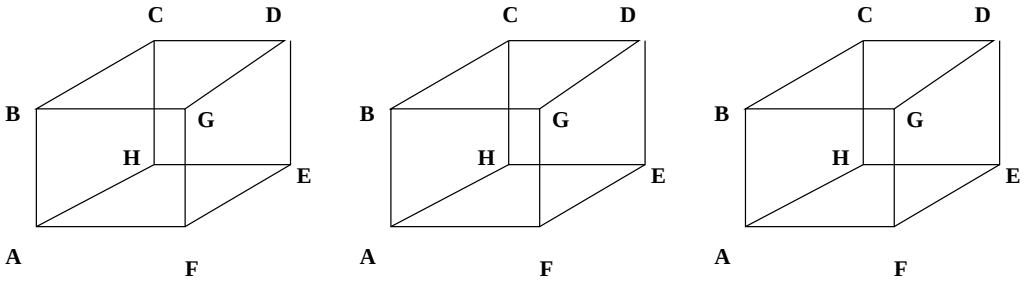


Figure 12.9: See the related Exercise 199. The *Necker Cube* has multiple interpretations: a human staring at one of these figures typically experiences changing interpretations. The interpretations of the two fork junctions in the center flip flop between the image of a front corner and the image of a back corner.

Exercise 200

Apply the interpretation tree procedure of Chapter 11 to the line drawings in Figure 12.7. Show any correct labelings resulting from completed paths.

12.3 3D Cues Available in 2D Images

An image is a 2D projection of the world. However, anyone who appreciates art or the movies knows that a 2D image can evoke rich 3D perceptions. There are many cues used to make 3D interpretations of 2D imagery.

Several depth cues can be seen in the image of Figure 12.10. Two sleeping persons occlude the bench, which occludes a post, which occludes an intricate railing, which occludes the trees, which occlude the buildings with the spires, which occlude the sky. The sun is off to the far right as is indicated by the shadow of the rightmost lamp post and its brighter right side. Also, the unseen railing at the right casts its intricate shadow on the planks, giving the false appearance of a tiled patio. The texture on the ground indicates a planar surface and the shrinking of the texture indicates the gradual regress of the ground away from the viewer. The orientation of the building wall at the left is obvious to a human interpreter from the orientation of the edges on it. The image of the railing tapers from right to left giving a strong cue that its depth is receding in 3D. Similarly, the boards of the bench taper from left to right. The images of the lamp post and the people are much larger than the images of the spires, indicating that the spires are far away.

96 DEFINITION **Interposition** *occurs when one object occludes another object, thus indicating that the occluding object is closer to the viewer than the occluded object.*

Object interposition gives very strong cues in interpreting the image of Figure 12.10 as discussed above. Clearly, the bench is closer than the post that it occludes, and the lamp post is closer than the railing. Recognition of individual objects may help in using this cue; however, it is not necessary. 'T' junctions formed in the image contours give a strong local cue. See Figure 12.11. Note that in the edge image at the right of Figure 12.10 the building edge is the bar of a 'T' junction formed with the top of the bench and the railing forms the bar of a 'T' junction with the right edge of the lamp post. A matching pair of 'T' junctions

Exercise 201

Apply the relaxation labeling procedure of Chapter 11 to the line drawings in Figure 12.7. If the label set for any edge becomes NULL, then there is no consistent interpretation. If any edges have more than one label in the final label set, then there are ambiguous interpretations as far as this algorithm is concerned. In such cases, use the multiple labels on the line drawing and verify which are actually realizable.

Figure 12.10: In Quebec City above the cliff above the great St Lawrence River: (left) image showing many depth cues, (right) result of Roberts edge operator followed by thresholding to pass 10% of the pixels.

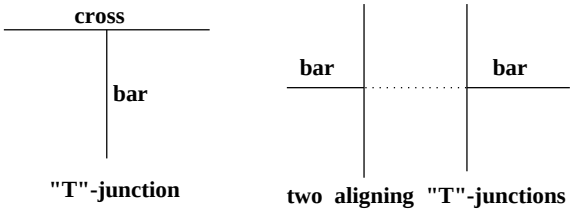


Figure 12.11: "T"-junctions indicate occlusion of one object by another: the inference is that the edge of the cross corresponds to the occluding object, while the edge of the bar corresponds to the occluded object. Two aligning "T"-junctions provide much stronger evidence of occlusion than just one.

Exercise 202

Find all 'T' junctions in the line segments of the box in Figure 12.3. Does each truly indicate occlusion of one surface by another?

is an even stronger cue because it indicates a continuing object passing behind another. This edge image is difficult as is often the case with outdoor scenes: for simpler situations, consider the next exercise. **Interposition of recognized objects or surfaces can be used to compute the relative depth of these objects.**

97 DEFINITION **Perspective scaling** *indicates that the distance to an object is inversely proportional to its image size. The term "scaling" is reserved for comparing object dimensions that are parallel to the image plane.*

Once we recognize the spires in Figure 12.10, we know they are quite distant because their image size is small. The vertical elements of the railing decrease in size as we see them recede in distance from right to left. Similarly, when we look down from a tall building to the street below, our height is indicated by how small the people and cars appear. **The size of an object recognized in the image can be used to compute the depth of that object in 3D.**

98 DEFINITION **Forshortening** *of an object's image is due to viewing the object at an acute angle to its axis and gives another strong cue of how the 2D view relates to the 3D object.*

As we look at the bench in Figure 12.10 and the people on it, their image length is shorter than it would be were the bench at a consistent closest distance to the viewer. Similarly, the verticle elements of the railing get closer together in the image as they get further away in the scene. This forshortening would not occur were we to look perpendicularly at the railing. *Texture gradient* is a related phenomena. Elements of a texture are subject to scaling and forshortening, so these changes give a viewer information about the distance and orientation of a surface containing texture. This effect is obvious when we look up at a brick building, along a tiled floor or railroad track, or out over a field of corn or a stadium crowd. Figure 12.12 demonstrates this. Texture gradients also tell us something about the shape of our friends' bodies when they wear clothes with regular textured patterns. A sketch of a simple situation creating a texture gradient is given in Figure 12.13: the texels, or images of the dots, move closer and closer together toward the center of the image corresponding to the increasing distance in 3D. Figure 12.14 shows how a texture is formed when objects in a scene are illuminated by a regular grid of light stripes. This artificial (structured) lighting not only gives humans more information about the shape of the surfaces, but also allows automatic computation of surface normal or even depth as the next chapter will show. Change of texture in an image can be used to compute the orientation of the 3D surface yielding that texture.

99 DEFINITION **Texture gradient** *is the change of image texture (measured or perceived) along some direction in the image, often corresponding to either a change in distance or surface orientation in the 3D world containing the objects creating the texture.*

Regular textured surfaces in 3D viewed nonfrontally create texture gradients in the image, but the reverse may not be true. Certainly, artists create the illusion of 3D surfaces by creating texture gradients on a single 2D sheet of paper.

100 DEFINITION **Motion parallax** *gives a moving observer information about the depth to objects, as even stationary objects appear to move relative to each other: the images of closer objects will move faster than the images of distant objects.*

Figure 12.12: Image of a cornfield shows multiple textures (corn plants and rows of corn plants) and texture gradients. Texture becomes more dense from bottom to top in the image because each square centimeter of image contains more corn leaves. Photo courtesy of John Gerrish.

Although motion parallax is the result of viewer motion, a similar effect results if the viewer is stationary and the objects are moving. Figure 12.13 relates the several effects under discussion to the perspective viewing projection. When we walk down a street (assume one eye closed), the image of objects we pass, such as trash cans or doorways, move much faster across our retina than do the images of the same kind of objects one block ahead of us. When driving in a car, oncoming vehicles in the distance present stable images which ultimately speed up to pass rapidly by our side window. Similarly, the images of cars passing by a standing person change much faster than the images of cars one block away. Motion parallax is related to scaling and forshortening by the same mathematics of perspective projection.

There are even more 3D cues available from single 2D images than what we have discussed. For example, distant objects may appear slightly more blueish than closer objects. Or, their images may be less crisp due to scattering effects of the air in between these objects and the viewer. Also, depth cues can be obtained from focusing, as will be discussed in Chapter 13. Moreover, we have not discussed some other assumptions about the real world; for example, we have not assumed a ground plane or a world with gravity defining a special vertical direction, two conditions under which the human visual system evolved.

Exercise 203

Examine a pencil with one eye closed. Keep it parallel to the line between your eyeballs and move it from your face to arms length. Is the change in image size due to scaling or forshortening or both? Now hold the pencil at its center and maintain a fixed distance between your eye and its center: rotate the pencil about its center and observe the change in its image. Is the change due to scaling or forshortening or both? Write an approximate trigonometic formula for image size as a function of rotation angle.

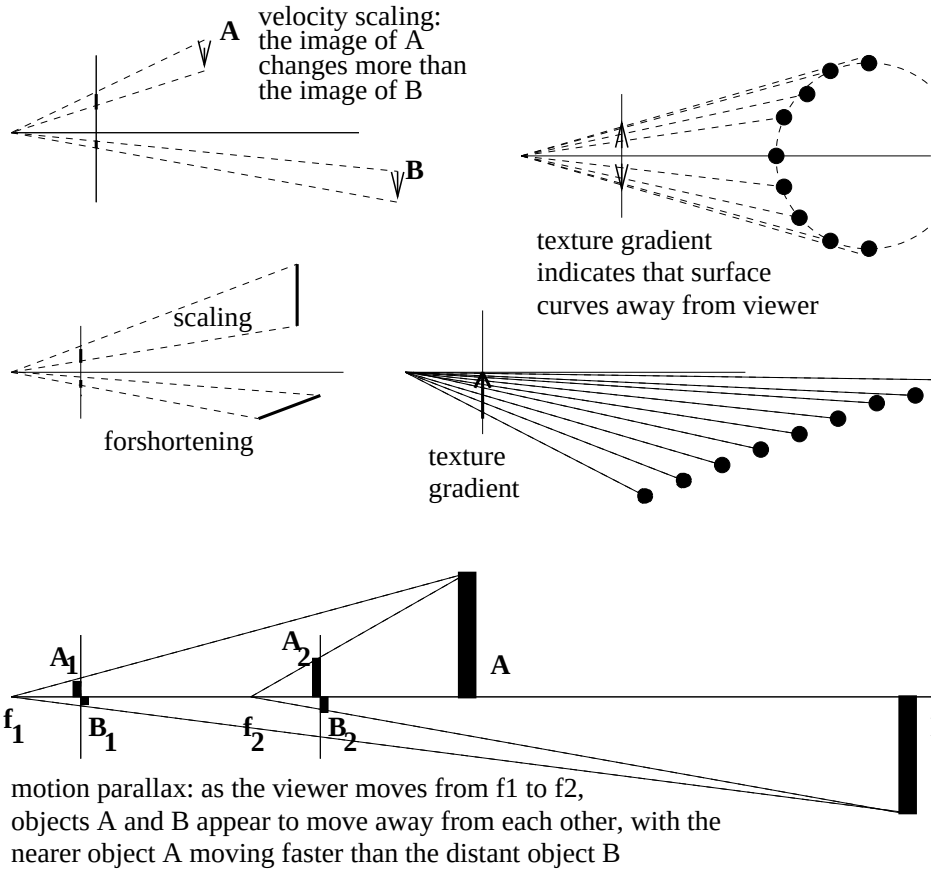


Figure 12.13: Sketches of the effects of scaling, forshortening, texture gradient and motion parallax. In each case, the front image plane is represented by a vertical line segment; objects are toward the right.

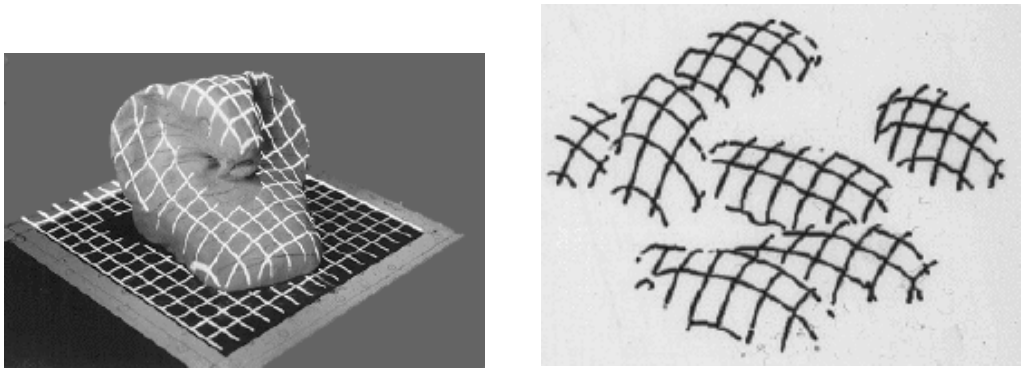


Figure 12.14: The texture formed by light stripes projected onto objects reveals their 3D surface shape: (left) a grid of light projected on a smooth sculpture on a planar background; (right) stripes on objects with planar background removed – what are the objects? Images courtesy of Gongzhu Hu.

Exercise 204

Hold a finger vertically closely in front of your nose and alternately open one eye only for 2 seconds. Observe the apparent motion of your finger (which should not be moving). Move your finger further away and repeat. Move your finger to arms length and repeat again. (It might help to line up your finger tip with a door knob or other object much farther away.) Describe the amount of apparent motion relative to the distance of finger to nose.

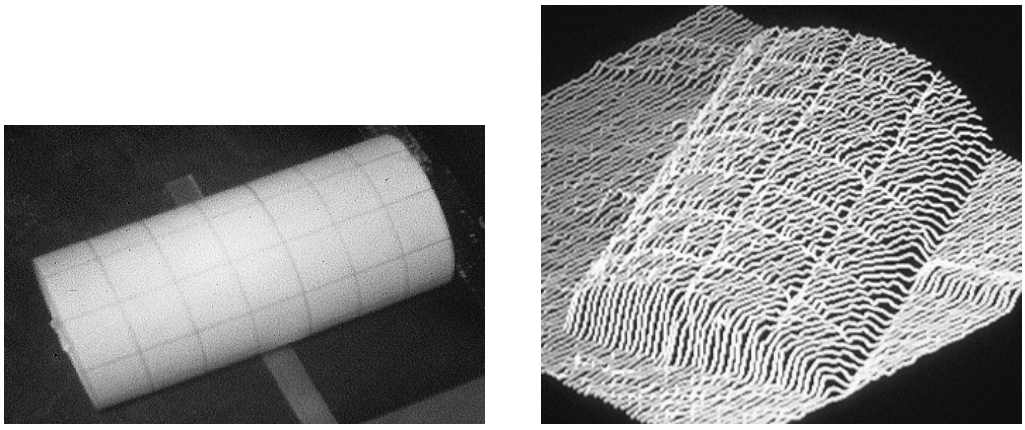


Figure 12.15: (Left) Image of a carefully illuminated cylinder formed by wrapping a gridded paper around a can and (right) a 3D plot of the intensity function from a slightly different viewpoint. Note how the shape of cylinder is well represented by the intensities.

12.4 Other Phenomena

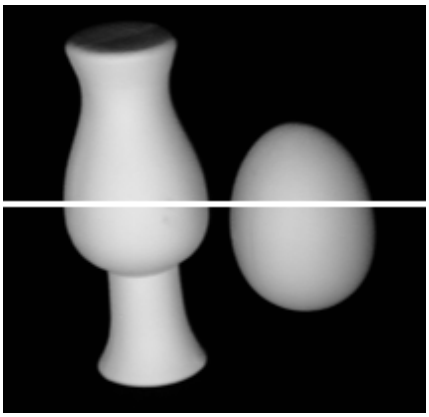
In Chapter 10, we discussed principles, such as the Gestalt principles, for grouping image features to make larger structures. These principles are often fruitful in inverting the image to obtain a 3D interpretation — and sometimes are deceiving, in the sense that they stimulate incorrect interpretations for some sets of conditions. Some additional important phenomena for interpreting 3D structure from 2D image features are briefly discussed below.

12.4.1 Shape from X

The 1980's saw a flurry of work on computational models for computing surface shape from different image properties. The research usually concentrated on using a single image property rather than combinations of them. Some of the mathematical models are examined in Chapter 13, while the phenomena used are discussed in more detail below. Here we just introduce the property **X** used as a 3D shape cue.

Shape from Shading:

Use of shading is taught in art class as an important cue to convey 3D shape in a 2D image. Smooth objects, such as an apple, often present a highlight at points where a reception from the light source makes equal angles with reflection toward the viewer. At the



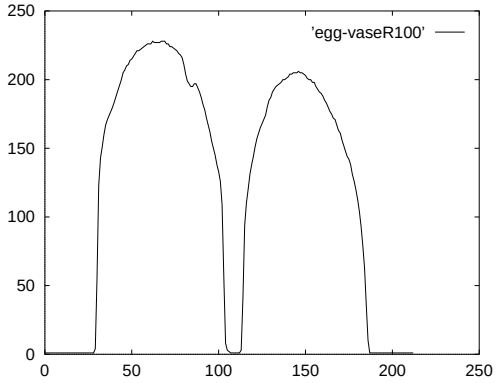


Figure 12.16: Intensity image of smooth buff objects — a vase and an egg — and a plot of intensities across the highlighted row. Note how the intensities are closely related to the object shape. Original image courtesy of Deborah Trytten.

same time, smooth objects get increasingly darker as the surface normal becomes perpendicular to rays of illumination. Planar surfaces tend to have a homogeneous appearance in the image with intensity proportional to the angle made between the normal to the plane and the rays of illumination. Computational formulas have been developed for computing surface normal from image intensity; however, most methods require calibration data in order to model the relationship between intensity and normal and some methods require multiple cameras. Recall that computational procedures must in general use a model formula relating several parameters – the reflected energy, the incident energy, the direction of illumination, the direction of reflection, and the orientation and reflectance of the surface element. With so many parameters, SFS can only be expected to work well by itself in highly controlled environments. Figure 12.15 shows an image of a uniform cylinder with a grid of lines illuminated from a single direction, while Figure 12.16 shows two images of buff objects yielding good shading information for perceiving shape.

Shape from Texture:

Whenever texture is assumed to lie on a single 3D surface and to be uniform, the notion of texture gradient in 2D can be used to compute the 3D orientation of the surface. The concept of texture gradient has already been described. Figure 12.18 shows a uniform texture on a 3D surface viewed at an angle so that a texture gradient is formed in the 2D image. There are two angles specially defined to relate the orientation of the surface to the viewer.

101 DEFINITION *The* **tilt** *of a planar surface is defined as the direction of the surface normal projected in the image. The* **slant** *of a surface is defined to be the angle made between the surface normal and the line of sight. See Figure 12.18*

Consider a person standing erect and looking ahead in a flat field of wheat. Assuming that the head is vertical, the tilt of the field is 90 degrees. If the person is looking far away, then the slant is close to 90 degrees; if the person is looking just beyond her feet, then the

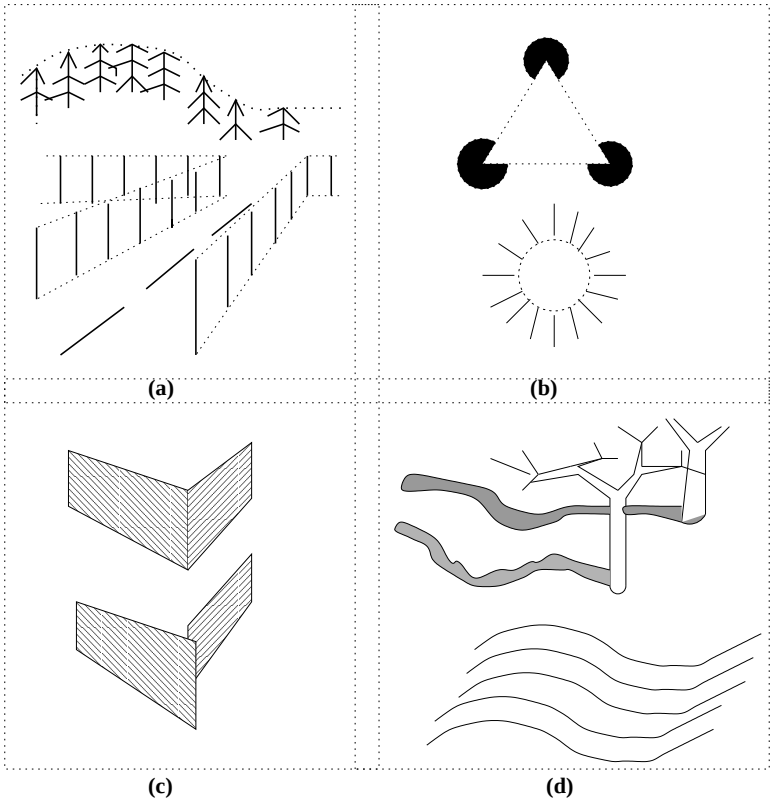


Figure 12.17: Other cues for inferring 3D from 2D image features. (a) Virtual lines and curves can be formed by grouping of similar features. (b) Virtual boundaries can deceive humans into perceiving interposing objects with intensity different from the background. (c) Alignments in 2D usually, but not always, indicate alignment in 3D. (d) 2D image curves induce perception of 3D surface shape.

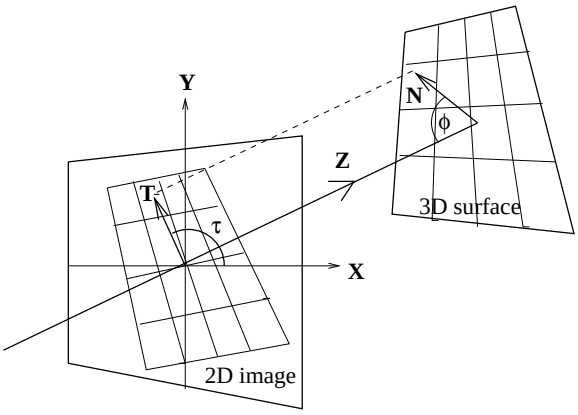


Figure 12.18: Tilt and slant of a surface are determined from the orientation of the surface normal **N** relative to the viewing coordinate system. *Tilt* τ is the direction of **N** projected into the image (**T**). *Slant* ϕ is the angle between **N** and the line of sight.

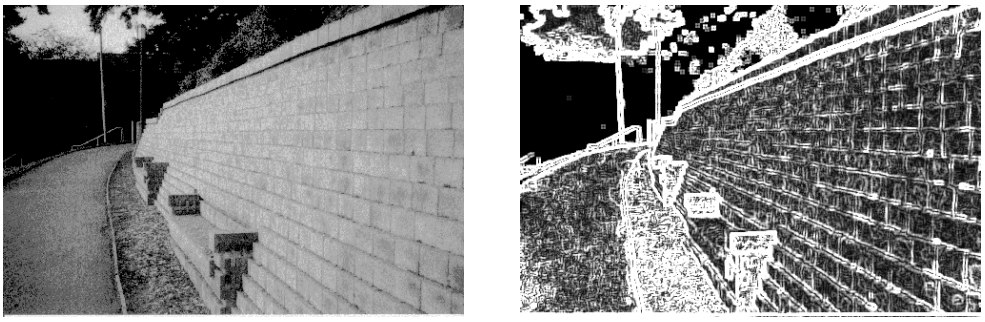


Figure 12.19: (Left) Image containing many textures and (right) result of 5x5 Prewitt edge operator. The walk has tilt of 90 degrees and slant of about 75 degrees: the brick wall has tilt of about 170 degrees and slant of about 70 degrees.

slant is close to zero. Now, if the person rotates her head 45 degrees to the left, then the tilt of the field becomes 45 degrees, while a 45 degree tilt of the head toward the right would produce a tilt of 135 degrees for the field surface. Figure 12.19 is a picture containing two major planar surfaces, a walkway on the ground plane and a terrace wall. The walkway has tilt of 90 degrees and slant of about 75 degrees. (The path inclines 15 degrees uphill.) The terrace wall has a tilt of about 170 degrees and slant of about 70 degrees. The concepts of tilt and slant apply to any surfaces, not just those near the *ground plane*; for example, outside or inside walls of buildings, the faces of boxes or trucks. In fact, these concepts apply to the elements of curved surfaces as well; however, the changing surface normal makes it more difficult to compute the texture gradient in the image.

Shape from Boundary:

Humans infer 3D object shape from the shape of the 2D boundaries in the image. Given an ellipse in the image, immediate 3D interpretations are *disk* or *sphere*. If the circle has

Exercise 205

(a) Give the tilt and slant for each of the four faces of the object in Figure 12.6. (b) Do the same for the faces of the objects in Figure 12.1.

uniform shading or texture the disk will be favored, if the shading or texture changes appropriately toward the boundary, the sphere will be favored. Cartoons and other line drawings often have no shading or texture, yet humans can derive 3D shape descriptions from them.

Computational methods have been used to compute surface normals for points within a region enclosed by a smooth curve. Consider the simple case of a circle: the smoothness assumption means that in 3D, the surface normals on the object limb are perpendicular to both the line of sight and the circular cross section seen in the image. This allows us to assign a unique normal to the boundary points in the image. These normals are in opposite directions for boundary points that are the endpoints of diameters of the circle. We can then interpolate smooth changes to the surface normals across an entire diameter, making sure that the middle pixel has a normal pointing right at the viewer. Surely, we need to make an additional assumption to do this, because if we were looking at the end of an elipse, the surface would be different from a sphere. Moreover, we could be looking at the *inside* of a half spherical shell! Assumptions can be used to produce a unique surface, which may be the wrong one. Shading information can be used to constrain the propagation of surface normals. Shading can differentiate between an egg and a ball, but maybe not between the outside of a ball and the inside of a ball.

Exercise 206

Find a cartoon showing a smooth human or animal figure. (a) Is there any shading, shadows, or texture to help with perception of 3D object shape? If not, add some yourself by assuming a light bulb at the front top right. (b) Trace the object boundary onto a clean sheet of paper. Assign surface normals to 20 or so points within the figure boundary to represent the object shape as it would be represented in an intrinsic image.

12.4.2 Vanishing Points

The perspective projection distorts parallel lines in interesting ways. Artists and draftspersons have used such knowledge in visual communication for centuries. Figure 12.20 shows two well-known phenomena. First, a 3D line skew to the optical axis will appear to vanish at a point in the 2D image called the *vanishing point*. Secondly, a group of parallel lines will have the same vanishing point as shown in the figure. These effects are easy to show from the algebraic model of projective projection. *Vanishing lines* are formed by the vanishing points from different groups of lines parallel to the same plane. In particular, a *horizon line* is formed from the vanishing points of different groups of parallel lines on the ground plane. In Figure 12.20, points $\mathbf{V_1}$ and $\mathbf{V_3}$ form a horizon for the ground plane formed by the rectangular tiled surface. Note that the independent bundle of three parallel lines (highway) vanishes at point $\mathbf{V_2}$ which is on the same horizon line formed by the rectangular texture. Using these properties of perspective, camera models can be deduced from imagery taken from ordinary uncalibrated cameras. Recently, systems have been constructed using these principles that can build 3D models of scenes from an ordinary video taken from several

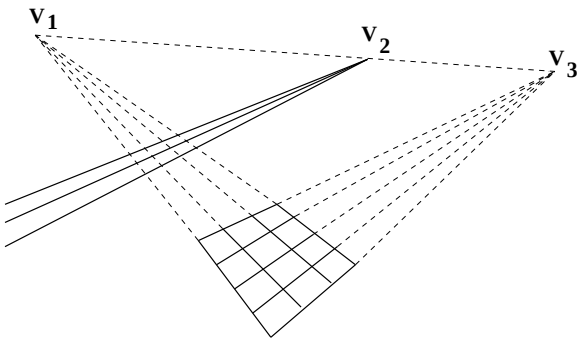


Figure 12.20: Under perspective projection, lines in 3D slanted with respect to the optical axis appear to vanish at some point in the 2D image, and parallel lines appear to intersect at the same *vanishing point*.

viewpoints in the scene.

12.4.3 Depth from Focus

A single camera can be used to compute the depth to the surface imaged at a pixel as can the human eye. Muscles in the human eye change its shape and hence the effective focal length causing the eye to focus on an object of attention. By bringing an object, or object edges, into focus, the sensor obtains information on the range to that object. Devices operating on this principle have been built, including the control for automatic focus cameras. Being brief here, we can imagine that the focal length of the camera is smoothly changed over a range of values; for each value of f an edge detector is applied to the resulting image. For each pixel, the value of f which produces the sharpest edge is stored and then used to determine the range to the 3D surface point imaged at that pixel. Many image points will not result from a contrasting neighborhood in 3D and hence will not produce usable edge sharpness values. Lenses with short focal length, for example, $f < 8mm$, are known to have a good *depth of field*, which means that an object remains in sharp focus over a wide range from the camera. Short focal lengths would not be good for determining an accurate range from focus; longer focal lengths do a better job. In the next chapter, we look at how this follows from the lens equation from physics.

Use of Intensity/Shadows

It has already been mentioned that structured illumination is useful for producing features on otherwise uniform surfaces. Shadows can be helpful in a similar manner. Humans and machines can infer the existence and shape of a surface from any pattern on it. Consider the bottom right example in Figure 12.17. The pattern of curves at the bottom stimulate a 3D interpretation of an undulating surface. The shadows from the trees on snow-covered terrain are helpful to skiers who can easily lose their balance from even a six inch mistake in judging ground elevation. A similar situation is shown in Figure 12.14, where the pattern of projected light stripes reveals the ellipsoidal shape of some potatoes.

12.4.4 Motion Phenomena

Motion parallax has already been considered. When a moving visual sensor pursues an object in 3D, points on that object appear to expand in the 2D image as the sensor closes in on the object. (Points on that object would appear to *contract*, not expand, if that object were escaping faster than the pursuer.) The point which is the center of pursuit is called the *focus of expansion*. The same phenomena results if the object moves toward the sensor: in this case, the rapid expansion of the object image is called *looming*. Chapter 9 treated these ideas in terms of optical flow. Quantitative methods exist to relate the image flow to the distance and speed of the objects or pursuer.

12.4.5 Boundaries and Virtual Lines

Boundaries or curves can be *virtual*, as shown in Figure 12.17. At the top left, image curves are formed by the ends of the fence posts, the tips of the trees, and the painted dashes on the highway. The top right shows two famous examples from psychology: humans see a brighter triangular surface occluding three dark circles, and a brighter circular surface occluding a set of rays radiating from a central point. It has been hypothesized that once the human visual system perceives (incorrectly) that there is an interposing object, it then must reject the interpretation that that object accidentally has the same reflectance as the background it occludes. So strong is this perception that the dotted virtual curves provided in Figure 12.17 need not be provided to a human at all. A machine vision system will not be fooled into perceiving that the central region is brighter than the background because it has objective pixel intensities.

102 DEFINITION **Virtual lines** *or curves are formed by a compelling grouping of similar points or objects along an image line or curve.*

Exercise 207

Carefully create two separate white cards containing the illusory figures at the top right of Figure 12.17. Show each to 5 human subjects to determine if they do see a brighter central region. You must not ask them this question directly: instead, just ask them general questions to get them to describe what they perceive. Ask *What objects do you perceive?* Ask *Please describe them in terms of their shape and color.* Summarize your results.

12.4.6 Alignments are Non-Accidental

Humans tend to reject interpretations that imply *accidental alignments* of objects in space or of objects and the viewer. Instead, alignments in the 2D image are usually assumed to have some cause in 3D. For example, a human viewing the two quadrilateral regions at the top of the panel in the lower left of Figure 12.17 will infer that two [rectangular] surfaces meet in 3D at the image edge, which is perceived as a crease due to the forshortening cues. The bottom of the panel shows how the image might appear after a small change in viewpoint: the fork and arrow junctions from the top have become **T**-junctions, and the perception is now of one surface occluding another. Perception of virtual curves is another form of this same principle. Actually, all of the four panels of Figure 12.17 could be included under the

same principle. As proposed in Irving Rock's 1983 treatise, the human visual system seems to accept the simplest hypothesis that explains the image data. (This viewpoint explains many experiments and makes vision similar to reasoning. However, it also seems to contradict some experimental data and may make vision programming very difficult.)

Some of the heuristics that have been proposed for image interpretation follow. None of these give correct interpretations in all cases: counterexamples are easy to find. We use the term *edge* here also to mean crease, mark, or shadow in 3D in addition to its 2D meaning.

- From a straight edge in the image, infer a straight edge in 3D.
- From edges forming a junction in the 2D image, infer edges forming a corner in 3D. (More generally, from coincidence in 2D infer coincidence in 3D.)
- From similar objects on a curve in 2D, infer similar objects on a curve in 3D.
- From a 2D polygonal region, infer a polygonal face in 3D.
- From a smooth curved boundary in 2D, infer a smooth object in 3D.
- Symmetric regions in 2D correspond to symmetric objects in 3D.

12.5 The Perspective Imaging Model

We now derive the algebraic model for perspective imaging. The derivation is rather simple when done by relating points in the camera coordinate frame **C** to the real image coordinate frame **R**. First, consider the simple 1D case shown in Figure 12.21, which is an adequate model for problems such as imaging a flat surface of earth from an airplane looking directly downward. The sensor observes a point **B** which projects to image point **E**. The center of the sensor coordinate system is point **O** and the distance from **O** to **B** measured along the optical axis from **O** to **A** is z_c. Point **B** images a distance x_i from the image center. f is the focal length. Using similar triangles, we obtain equation 12.1. This says that the real 2D image coordinate (or size) is just the 3D coordinate (or size) scaled by the ratio of the focal length to the distance. Provided that all points being imaged lie on the same plane at the same distance from the sensor, the image is just a scaled down version of the 2D world. This model is applicable to real applications such as analyzing microscope or areal images or scanned documents.

$$x_i/f \;=\; x_c/z_c \;\;\texttt{or}\;\; x_i \;=\; (f/z_c)\,x_c \tag{12.1}$$

It is convenient to think of a *front image plane* rather than the actual image plane so that the image is not inverted in direction relative to the world. The front image plane is an abstract image plane f units on the world side of the optical center: objects on the front image plane have the same proportions as those on the actual image plane and the same sense of direction as those in the real world. Points **C** and **D** are the front image plane points corresponding to points **F** and **E** on the real image plane. The perspective imaging equation holds for points on the front image plane. From now on, the front image plane will be used for our illustrations.

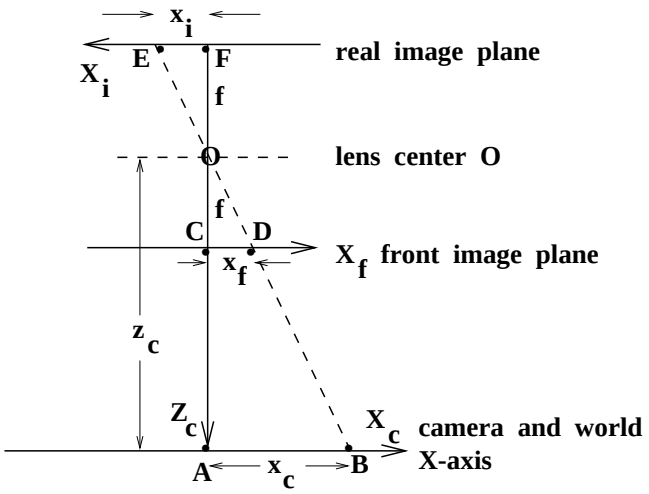


Figure 12.21: Simple perspective model with actual and front image planes. Object size in the world x_c is related to its image size x_i via similar triangles: $x_i/f = x_c/z_c$.

The case of perspective projection from 3D to 2D is shown in Figure 12.22 and modeled by the Equations 12.2. The equations for both x and y dimensions are obtained using the same derivation from similar triangles as in the 1D case. Note that, as a projection of 3D to 2D, this is a many-to-one mapping. All 3D points along the same ray from an image point out into the 3D space image to the same 2D image point and thus a great deal of 3D information is lost in the process. Equation 12.2 provides the algebraic model which computer algorithms must use to model the set of all 3D points on the ray from image point (x_i, y_i) out into the world. This algebra is of fundamental importance in the 3D work of the text. Before leaving this topic, we emphasize that the simple Equation 12.2 only relates points in the 3D camera frame to points in the 2D real image frame. Relating points from object coordinate frames or a world coordinate frame requires the algebra of transformations, which we will take up in Chapter 13. If the camera views planar material a constant distance $z_c = c_1$ away, then the image is a simple scaling of that plane. Setting $c_2 = f/c_1$ yields the simple relations $x_i = c_2 x_c$ and $y_i = c_2 y_c$. Thus we can simply work in image coordinates, knowing that the image is a scaled version of the world.

$$\begin{aligned} x_i/f &= x_c/z_c \;\; or \;\; x_i = (f/z_c)\, x_c \\ y_i/f &= y_c/z_c \;\; or \;\; y_i = (f/z_c)\, y_c \end{aligned} \tag{12.2}$$

Exercise 208 uniform scaling property

A camera looks vertically down at a table so that the image plane is parallel to the table top (similar to a photo enlarging station) as in Figure 12.21. Prove that the image of a 1 inch long nail (line segment) has the same length regardless of where the nail is placed on the table within the FOV.

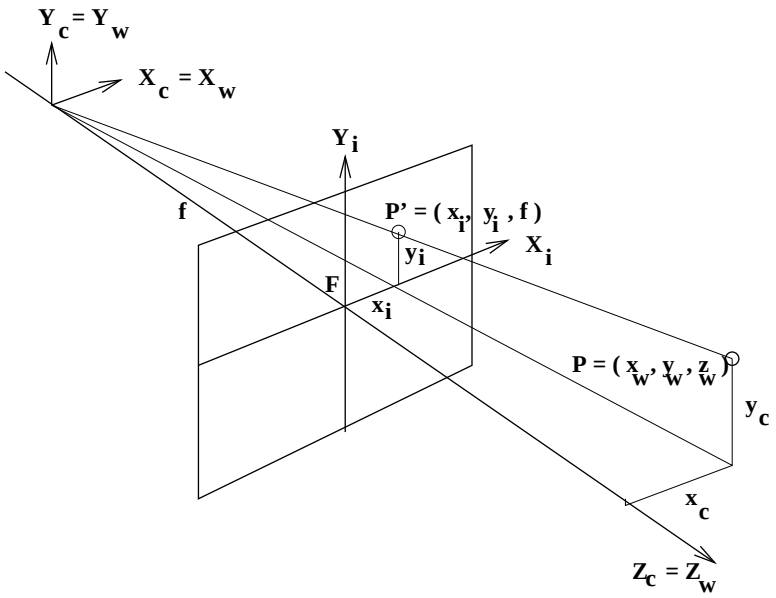


Figure 12.22: General model of perspective projection to a 2D image.

12.6 Depth Perception from Stereo

Only simple geometry and algebra is needed to understand how 3D points can be located in space using a stereo sensor as is shown in Figure 12.24. Assume that two cameras, or eyes, are carefully aligned so that their X-axes are collinear and their Y-axis and Z-axis are parallel. The Y-axis is perpendicular to the page and is not actually used in our derivations. The origin, or center of projection, of the right camera is offset by $\mathbf{b}$, which is the *baseline* of the stereo system. The system observes some object point $\mathbf{P}$ in the left image point $\mathbf{P_l}$ and the right image point $\mathbf{P_r}$. Geometrically, we know that the point $\mathbf{P}$ must be located at the intersection of the ray $\mathbf{LP_l}$ and the ray $\mathbf{RP_r}$.

From similar triangles, Equations 12.3 are obtained.

$$\begin{aligned} z/f &= x/x_l \\ z/f &= (x-b)/x_r \\ z/f &= y/y_l = y/y_r \end{aligned} \tag{12.3}$$

Note that by construction the image coordinates y_l and y_r may be assumed to be identical. A few substitutions and manipulations yields a solution for the two unknown coordinates x and z of the point $\mathbf{P}$.

$$\begin{aligned} z &= fb/(x_l - x_r) = fb/d \\ x &= x_l z/f = b + x_r z/f \\ y &= y_l z/f = y_r z/f \end{aligned} \tag{12.4}$$

In solving for the depth of point $\mathbf{P}$, we have introduced the notion of *disparity* d in Equations 12.4, which is the difference between the image coordinates x_l and x_r in the left and

Exercise 209 a vision-guided tractor

See Figure 12.23. Suppose a forward-looking camera is used to guide the steering of a farm tractor and its application of weed control and fertilizer. The camera has a focal length of 100mm and is positioned 3000mm above the ground plane as shown in the figure. It has an angular field of view of 50 degrees and its optical axis makes an angle of 35 degrees with the ground plane. (a) What is the length of the FOV along the ground ahead of the tractor? (b) Assuming that plants are spaced 500mm apart, what is their spacing in the image when they are at the extremes of the FOV? (Image spacing is different for near and far plants.) (c) If plants are approximated as spheres 200mm in diameter and the image is 500 x 500 pixels, what is their image diameter in pixels? (Again, different answer for near and far plants.) (d) Will successive plants overlap each other in the image or will there be space in between them?

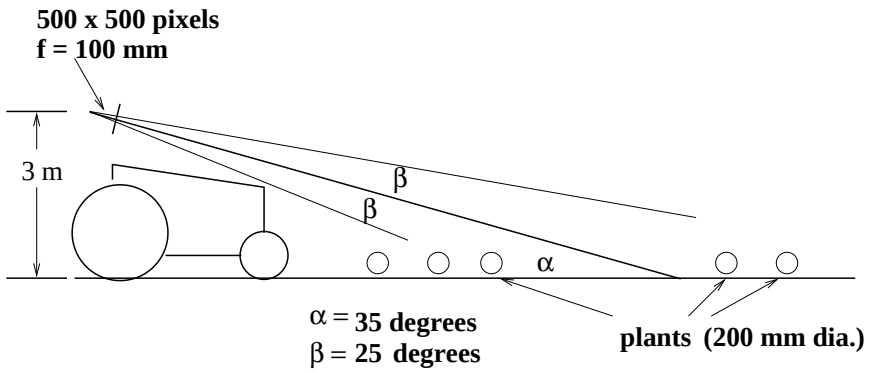


Figure 12.23: Sketch of camera on a vision-guided farm tractor: see Exercise 209.

right images. Solution of these equations yields all three coordinates completely locating point $\mathbf{P}$ in 3D space. Equation 12.4 clearly shows that the distance to point $\mathbf{P}$ increases as the disparity decreases and decreases as the disparity increases. Distance goes to infinity as disparity goes to zero. By construction of this simple stereo imaging system, there is no disparity in the two y image coordinates.

103 DEFINITION **Disparity** *refers to the difference in the image location of the same 3D point when projected under perspective to two different cameras.*

Figure 12.24 shows a single point $\mathbf{P}$ being located in 3D space so there is no problem identifying the matching image points $\mathbf{P_l}$ and $\mathbf{P_r}$. Determining these corresponding points can be very difficult for a real 3D scene containing many surface points because it is often unclear which point in the left image corresponds to which point in the right image. Consider a stereo pair of images from a cornfield as in Figure 12.12. There would be many similar edge points across the image rows. Often, stereo cameras are very precisely aligned so that the search for corresponding points can be constrained to the same rows of the two images. Although many constraints are known and used, problems still remain. One obvious

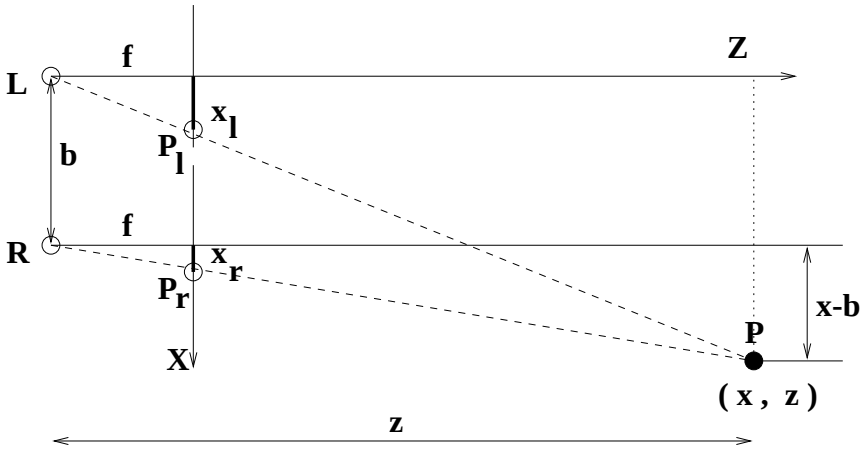


Figure 12.24: Geometric model for a simple stereo system: the sensor coordinate system is established in the left eye **L** (or camera) and the *baseline* is **b**; all quantities are measured relative to **L**, except for $\mathbf{x_r}$ which is measured relative to **R**.

example is the case where a point **P** is not visible in both images. While the busy texture of the cornfield presents problems in handling too many feature points, the opposite problem of having too few points is also common. Too few points are available for smooth objects with no texture, such as a marble statue or a snow-covered hill. In an industrial setting, it is easy to add artificial points using illumination as is shown in Figure 12.14. More details are given below.

Despite the difficulties mentioned above, research and development has produced several commercially available stereo systems. Some designs use more than two cameras. Some systems can produce depth images at nearly the frame rate of video cameras. Chapter 16 discusses use of stereo sensing at an ATM machine to identify people.

Exercise 210

Perform the following informal stereo experiment. (a) View a book frontally about 30 cm in front of your nose. Use one eye for two seconds, then the other eye, in succession. Do you observe the disparity of the point features, say the characters in the title, in the left and right images? (b) Repeat the experiment holding the book at arms length. Are the disparities larger or smaller? (c) Rotate the book significantly. Can you find an orientation such that the right eye sees the cover of the book but the left eye cannot?

Exercise 211 On the error in stereo computations

Assume that stereo cameras with baseline $b = 10cm$ and focal lengths $f = 2cm$ view a point at $P = (10cm, 1000cm)$. Refer to Figure 12.24: note that point P lies along the optical axis of the right camera. Suppose that due to various errors, the image coordinate x_l is 1% smaller than its true value, while the image coordinate x_r is perfect. What is the error in depth z, in centimeters, computed by Equations 12.4?

Stereo Displays

Stereo displays are generated by computer graphics systems in order to convey 3D shape to an interactive user. The graphics problem is the inverse of the computer vision problem: all the 3D surface points (x, y, z) are known and the system must create the left and right images. By rearranging the Equations 12.4 we arrive at Equations 12.5, which give formulas for computing image coordinates (x_l, y_l) and (x_r, y_r) for a given object point (x, y, z) and fixed baseline b and focal length f. Thus, given a computer model of an object, the graphics system generates two separate images. These two images are presented to the user in one of two ways: (a) one image is piped to the left eye and one is piped to the right eye using a special helmet or (b) the two images are presented alternately on a CRT using complimentary colors which the user views with different filters on each eye. There is an inexpensive third method if motion is not needed: humans can actually fuse a stereo pair of images printed side-by-side on plain paper. (For example, stare at the stereogram that is Figure 7 of the paper by Tanimoto (1998) cited in the references.)

$$\begin{aligned} x_l &= xf/z \\ x_r &= f(x-b)/z \\ y_l &= y_r = yf/z \end{aligned} \tag{12.5}$$

Chapter 15 discusses in more detail how stereo displays are used in virtual reality systems, which engage users to an increased degree as a result of the 3D realism. Also, they can be very useful in conveying to a radiologist the structure of 3D volumetric data from an MRI device.

12.6.1 Establishing Correspondences

The most difficult part of a stereo vision system is not the depth calculation, but the determination of the correspondences used in the depth calculation. If any correspondences are incorrect, they will produce incorrect depths, which can be just a little off or totally wrong. In this section we discuss the major methods used for finding correspondences and some helpful constraints.

Cross Correlation

The oldest technique for finding the correspondence between pixels of two images uses the cross-correlation operator described in Chapter 5. The assumption is that for a given point P_1 in image I_1 (the first image of a stereo pair), there will be a fixed region of image I_2 (the second image of the pair) in which the point P_2 that corresponds to P_1 must be found. The size of the region is determined by information about the camera setup used to take the images. In industrial vision tasks, this information is readily available from the camera parameters obtained by the calibration procedure (See Chapter 13). In remote sensing and other tasks, the information may have to be estimated from training images and ground truth. In any case, for pixel P_1 of image I_1, the selected region of I_2 is searched, applying the cross-correlation operator to the neighborhoods about P_1 and P_2. The pixel that maximizes the response of the cross correlation operator is deemed the best match to P_1 and

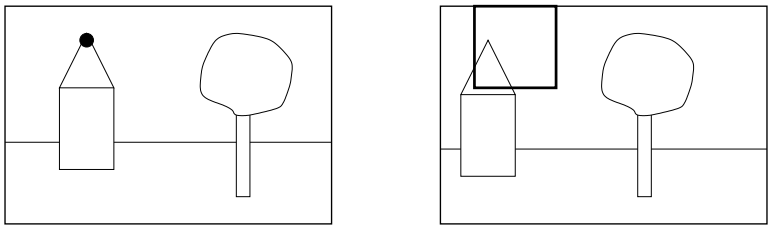


Figure 12.25: The cross-correlation technique for finding correspondences in a stereo pair of images.

used to find the depth at the corresponding 3D point. The cross-correlation technique has been used very successfully to find correspondences in satellite and aerial imagery. Figure 12.25 illustrates the cross-correlation technique. The black dot in image I_1 indicates a point whose correspondence is sought. The square region in image I_2 is the region to be searched for a match.

Symbolic Matching and Relational Constraints

A second common approach to finding correspondences is to look for a feature in one image that matches a feature in the other. Typical features used are junctions, line segments, or regions. This type of matching can use the consistent labeling formalism defined in Chapter 11. The part set P is the set of features of the first image I_1. The label set L is the set of features of the second image I_2. If features can have more than one type, then the label for a part must be of the same type as that part. (Note that 'T' junctions are often avoided since they usually are due to the occlusion of one edge by another and not to the structure of one 3D object.) Furthermore, the spatial relationships R_P that hold over P should be the same as the spatial relationships R_L that hold over L. For instance, if the features to be matched are junctions, as shown in Figure 12.26, then corresponding junctions should have the same types (an 'L' junction maps to another 'L' junction) and if two junctions are connected by a line segment in the first image ('L' and 'A' for example), the corresponding junctions should be connected by a line segment in the second image. If the features to be matched are line segments, such relationships as parallel and collinear can be used in matching. For region matching, the region adjacency relationship can be used.

This brings up one problem that can occur in any kind of stereo matching. Not every feature of the first image will be detected in the second. Some features are just not there, due to the viewpoint. Some features appear in one image, but are occluded in the other. Some features may be misdetected or just missed, and extraneous features may be found. So the symbolic matching procedure cannot look for a perfect consistent labeling, but instead must use some inexact version, either looking for a least-error mapping or applying continuous relaxation to achieve an approximate answer.

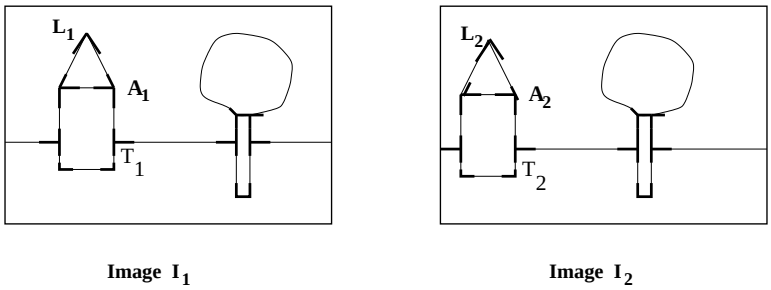


Figure 12.26: Symbolic matching technique for finding correspondences in a stereo pair of images. The 'L' and 'A' (arrow) junctions shown are potential matches. The 'T' junctions probably should be avoided since they usually result from occlusion rather than real features in 3D.

Once a mapping has been found from features of the first image to features of the second, we are not yet done. The correspondence of junctions produces a sparse depth map with the depth known only at a small set of points. The correspondence of line segments can lead to correspondences between their endpoints or, in some work, between their midpoints. The correspondences between regions still requires some extra work to decide which pixels inside the regions correspond. The sparse depth maps obtained can then be augmented through linear interpolation between the known values. As you can see, there is a lot of room for error in this process and that is probably why cross correlation is still widely used in practice, especially where the images are natural, rather than industrial, scenes.

The Epipolar Constraint

Stereo matching can be greatly simplified if the relative orientation of the cameras is known, as the two-dimensional search space for the point in one image that corresponds to a given point in a second image is reduced to a one-dimensional search by the so called *epipolar geometry* of the image pair. Figure 12.27 shows the epipolar geometry in the simple case where the two image planes are identical and are parallel to the baseline. In this case, given a point $P_1 = (x_1, y_1)$ in image I_1 , the corresponding point $P_2 = (x_2, y_2)$ in image I_2 is known to lie on the same scan line; that is, $y_1 = y_2$. We call this a *normal* image pair.

While the normal set up makes the geometry simple, it is not always possible to place cameras in this position, and it may not lead to big enough disparities to calculate accurate depth information. The general stereo setup has arbitrary positions and orientations of the cameras, each of which must view a significant subvolume of the object. Figure 12.28 shows the epipolar geometry in the general case.

104 DEFINITION *The plane that contains the 3D point P, the two optical centers (or cam-*

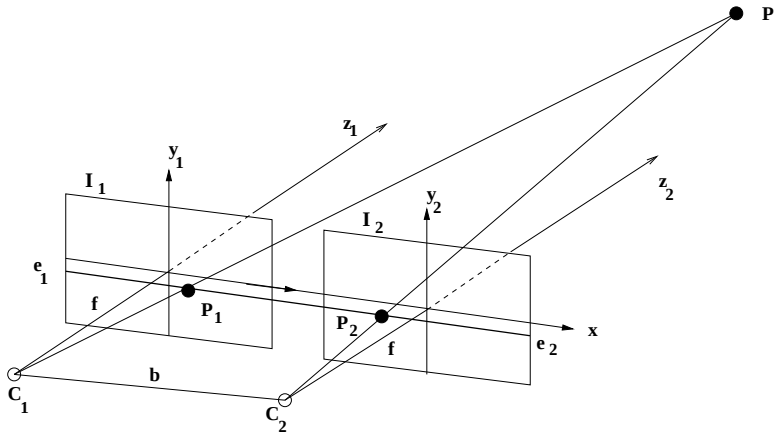


Figure 12.27: Epipolar geometry of a normal image pair: Point P in 3D projects to point P_1 on image I_1 and point P_2 on image I_2, which share the same image plane that is parallel to the baseline between the cameras. The optical axes are perpendicular to the baseline and parallel to each other.

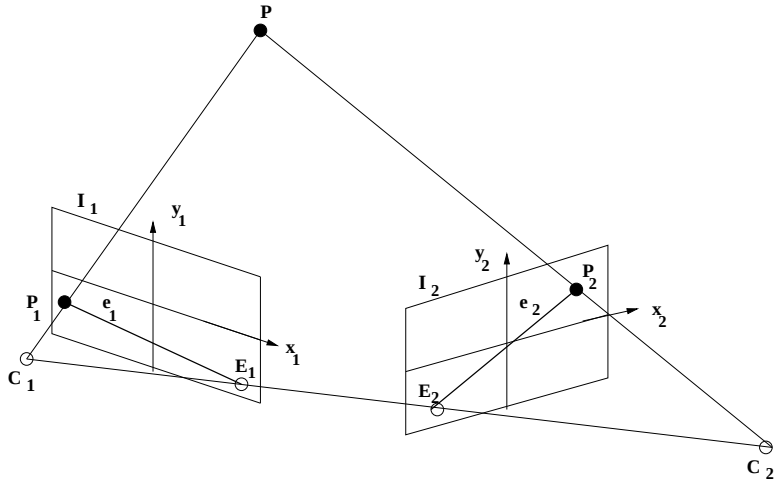


Figure 12.28: Epipolar geometry of a general image pair: Point P in 3D projects to point P_1 on image I_1 and point P_2 on image I_2; these image planes are not the same. The epipolar line that P_1 lies on in image I_1 is line e_1, and the corresponding epipolar line that P_2 lies on in image I_2 is line e_2. E_1 is the epipole of image I_1, and E_2 is the epipole of image I_2.

eras) C_1 and C_2, and the two image points P_1 and P_2 to which P projects is called the **epipolar plane**.

105 DEFINITION *The two lines e_1 and e_2 resulting from the intersection of the epipolar plane with the two image planes I_1 and I_2 are called* **epipolar lines**.

Given the point P_1 on epipolar line e_1 in image I_1 and knowing the relative orientations of the cameras (see Ch 13), it is possible to find the corresponding epipolar line e_2 in image I_2 on which the corresponding point P_2 must lie. If another point P_1' in image I_1 lies in a different epipolar plane from point P_1, it will lie on a different epipolar line.

106 DEFINITION *The* **epipole** *of an image of a stereo pair is the point at which all its epipolar lines intersect.*

Points E_1 and E_2 are the epipoles of images I_1 and I_2, respectively.

The Ordering Constraint

Given a pair of points in the scene and their corresponding projections in each of the two images, the ordering constraint states that if these points lie on a continuous surface in the scene, they will be ordered in the same way along the epipolar lines in each of the images. This constraint is more of a heuristic than the epipolar constraint, because we don't know at the time of matching whether or not two image points lie on the same 3D surface. Thus it can be helpful in finding potential matches, but it may cause correspondence errors if it is strictly applied.

Error versus Coverage

In designing a stereo system, there is a tradeoff between scene coverage and error in computing depth. When the baseline is short, small errors in location of image points P_1 and P_2 will propagate to larger errors in the computed depth of the 3D point P as can be inferred from the figures. Increasing the baseline improves accuracy. However, as the cameras move further apart, it becomes more likely that image point correspondences are lost due to increased effects of occlusion. It has been proposed that an angle of $\pi/4$ between optical axes is a good compromise.

12.7 * The Thin Lens Equation

The principle of the thin lens equation is shown in Figure 12.29. A ray from object point $\mathbf{P}$ parallel to the optical axis passes through the lens and focal point $\mathbf{F_i}$ in route to image point $\mathbf{p}'$. Other rays from $\mathbf{P}$ also reach $\mathbf{p}'$ because the lens is a light collector. A ray through the optical center $\mathbf{O}$ reaches $\mathbf{p}'$ along a straight path. A ray from image point $\mathbf{p}'$ parallel to the optical axis passes through the lens and second focal point $\mathbf{F_j}$.

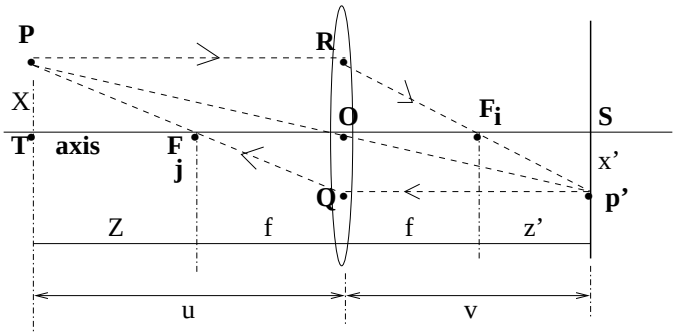


Figure 12.29: Principle of the thin lens. Ray from object point **P** parallel to the optical axis passes through the lens and focal point $\mathbf{F_i}$ in route to image point $\mathbf{p}'$. Ray from image point $\mathbf{p}'$ parallel to the optical axis passes through the lens and focal point $\mathbf{F_j}$.

The thin lens equation can be derived from the geometry in Figure 12.29. Since distance X is the same as the distance from **R** to **O**, similar triangles $\mathbf{ROF_i}$ and $\mathbf{Sp'F_i}$ give the following equation.

$$\frac{X}{f} = \frac{x'}{z'} \tag{12.6}$$

Using similar triangles **POT** and $\mathbf{p'OS}$ we obtain a second equation.

$$\frac{X}{f+Z} = \frac{x'}{f+z'} \tag{12.7}$$

Substituting the value of X from Equation 12.6 into Equation 12.7 yields

$$f^2 = Zz' \tag{12.8}$$

Substituting $u - f$ for Z and $v - f$ for z' yields

$$uv = f(u+v) \tag{12.9}$$

and finally dividing both sides by (uvf) yields the most common form of the lens equation, which relates the focal length to the object distance u from the lens center and the image distance v from the lens center.

$$\frac{1}{f} = \frac{1}{u} + \frac{1}{v} \tag{12.10}$$

Focus and Depth of Field

Assuming that a point **P** is in perfect focus as shown in Figure 12.29, the point will be out of focus if the image plane is moved, as shown in Figure 12.30. The lens equation, which held for v, is now violated for the new distance v'. Similarly, if the image plane remains constant but point **P** is moved, distance u changes and the lens equation is violated. In either case, instead of a crisp point on the image plane, the image of a world point spreads out into a circle of diameter b. We now relate the size of this circle to the resolution of the camera and its depth of field. In Chapter 13, we show how to search through image positions **S** to determine the depth u of a world point.

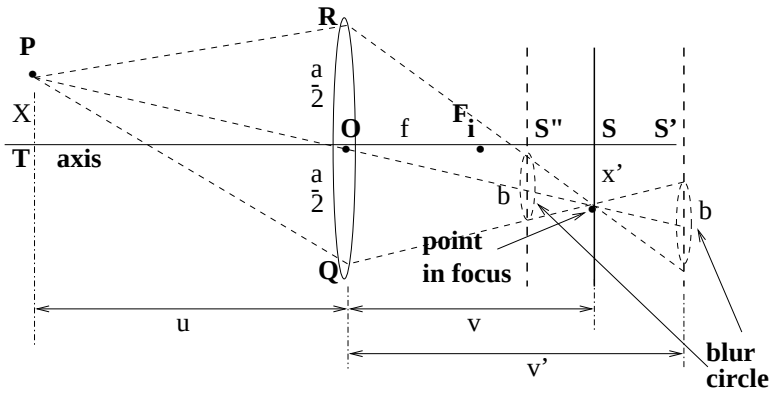


Figure 12.30: Point **P** will be out of focus if either the depth of the point or the position of the image plane violates the lens equation. If **S** is the image location where **P** is in focus, then **P** will be out of focus as the image plane is moved to $\mathbf{S}'$ or $\mathbf{S}''$. The image of **P** will be a disk of diameter b.

Assume that a blur b the size of a pixel can be tolerated. From this assumption, we will compute the nearest and farthest that a point **P** can be away from the camera and have a blur within this limit. We fix distance u as the nominal depth at which we want to sense; v as the ideal image distance according to the lens equation; a as the aperture of the lens; and f as the focal length used. The situation is sketched in Figure 12.30. Assuming that perfect focus is achieved for the above conditions, we now investigate how much u can be varied keeping the blur within the limit of b.

Using the extreme cases of v' in Figure 12.30, similar triangles yield

$$\begin{aligned} v' &= \frac{a+b}{a}v \quad : in \;\; case \;\; v' > v \\ v' &= \frac{a-b}{a}v \quad : in \;\; case \;\; v' < v \end{aligned} \tag{12.11}$$

Note that for $v' > v$, the lens equation shows that u' will be closer to the camera than u and for $v' < v$, u' will be farther. We compute the nearest point u_n that will result in the blur of b shown in the figure, using the lens equation several times to relate the parameters u, v, f and using the relation for $v' > v$ from Equation 12.11.

$$\begin{aligned} u_n &= \frac{fv'}{v'-f} = \frac{f\frac{(a+b)}{a}v}{\frac{(a+b)}{a}v - f} \\ &= \frac{f\;\frac{(a+b)}{a}\;\frac{uf}{(u-f)}}{\frac{(a+b)}{a}\;\frac{uf}{(u-f)} \;-\; f} \\ &= \frac{uf(a+b)}{af+bu} = \frac{u(a+b)}{a+\frac{bu}{f}} \end{aligned} \tag{12.12}$$

Similarly, by using $v' < v$ and the same steps given above, we obtain the location u_r of the far plane.

$$u_r = \frac{uf(a-b)}{af-bu} = \frac{u(a-b)}{a-\frac{bu}{f}} \tag{12.13}$$

107 DEFINITION *The* **depth of field** *is the difference between the far and near planes for the given imaging parameters and limiting blur b.*

Since $u > f$ in typical situations, one can see from the final expression in Equation 12.12 that $u_n < u$. Moreover, holding everything else constant, using a shorter focal length f will bring the near point closer to the camera. The same kind of argument shows that $u_r > u$ and that shortening f will push the far point farther from the camera. Thus, shorter focal length lenses have a larger depth of field than lenses with longer focal lengths. (Unfortunately, shorter focal length lenses often have more radial distortion.)

Relating Resolution to Blur

A CCD camera with ideal optics can at best resolve $n/2$ distinct lines across n rows of pixels, assuming that a spacing of one pixel between successive pairs of lines is necessary. A 512×512 CCD array can detect a grid of 256 dark lines separated by one-pixel wide rows of bright pixels. (If necessary, we can move the camera perpendicular to the optical axis slightly until the line pattern aligns optimally with the rows of pixels.) Blur larger than the pixel size will fuse all the lines into one grey image. The formulas given above enable one to engineer the imaging equipment for a given detection problem. Once it is known what features must be detected and what standoff is practical, one can then decide on a detector array and lens.

108 DEFINITION *The* **resolving power** *of a camera is defined as* $R_p = 1/(2\Delta)$ *in units of lines per inch (mm) where* Δ *is the pixel spacing in inches (mm).*

For example, if a CCD array 10mm square has 500×500 pixels, the resolving power is $1/(2 \times 2 \times 10^{-2} mm/line)$ or 25 lines per mm. If we assume that black/white film is comprised of silver halide molecules 5×10^{-3} mm apart, the resolving power is 100 lines per mm or 2500 lines per inch. The cones that sense color in the human eye are packed densely in an area called the *fovea*, perhaps with a spacing of $\Delta = 10^{-4}$ inches. This translates to a resolving power of 5×10^3 lines per inch on the retina. Using $20mm = 0.8in$ as the diameter of an eye, we can compute the subtended angle shown in Figure 12.31 as $\theta \approx sin(\theta) = 2\Delta/0.8in = 2.5 \times 10^{-4}$ radians, which is roughly one minute of arc. This means that a human should be able to detect a line made by a 0.5 mm lead pencil on a wall 2 meters away.

12.8 Concluding Discussion

This chapter examined a number of relations between 2D image structure and the surfaces and objects in the 3D world. Humans use many of these relations in combination to perceive the world and to navigate within it. Although we only took a careful look at the relationship of depth to stereo and focus, quantitative models have been formulated for many phenomena,

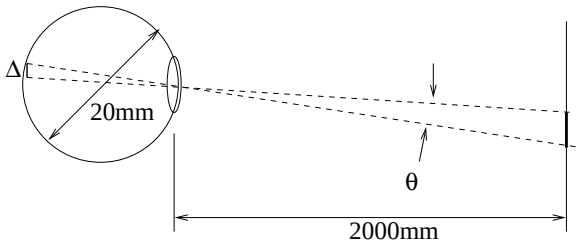


Figure 12.31: A small object images on the human retina.

including shape from shading and shape from texture. These models are important tools for artists, especially when computer graphics is used, to convey 3D structure using a 2D canvas or 2D graphics display. Chapter 13 shows how some of these methods can be employed for automatic recognition of 3D structure from 2D images. We caution the reader that some of these models are too fragile and inaccurate to be used by themselves except in controlled environments. Combination of computational methods to provide real-time vision for an outdoor navigating robot is still a difficult problem and the subject of much current research activity.

Exercise 212

Locate a painting of a plaza or acropolis in an art book and make a copy of it. Mark vanishing points and vanishing lines that the artist may have used in creating the picture.

12.9 References

The early book by visual psychologist J.J. Gibson (1950) is a classic in the study of visual information cues. Much of the work of computer vision researchers of the 1980s can be traced to roots in Gibson's book. Many experiments were influenced by the approach of David Marr (1982), whose "information processing paradigm" held that one should first isolate the information being used to make a decision or perception, second explore mathematical models and finally pursue possible implementations. Marr also believed that the human visual system actually constructed fairly complete descriptions of the surfaces of the scene, a belief that is less common today. The 1983 book by visual psychologist Irvin Rock contains a retrospective view of many years of experiments and reaches the conclusion that visual perception requires intelligent operations and shares many properties of reasoning: this book is both a resource for properties of human vision and also for the methodology used to study it. The notion of the intrinsic image was introduced by Barrow and Tenanbaum in 1978. Their proposal is roughly equivalent to the 2 1/2 D sketch proposed by David Marr and appearing in his 1982 book. Our treatment of the intrinsic image has been strongly influenced by the development in Chapter 3 of the text by Charniak and McDermott (1985).

Huffman 1971 and Clowes 1971 are jointly credited with discovery of the junction constraints in blocks world. The extension of that work by Waltz in 1975 to handle shadows and nontrihedral corners enlarged the junction catalogue to thousands of cases – probably too much for a consciously reasoning human to manipulate, but no trouble for a computer.

Waltz developed an efficient algorithm to discard possible line labels, which is offen dubbed "Waltz filtering". The 1977 AI text by Winston is a good source of details on the topic of shape interpretation in blocks world: it also contains a treatment of how the junction catalogues are derived.

The parallel relaxation approach which we gave here is derived from a large amount of similar work by Rosenfeld et al (1976), and others, some of which grew from Waltz's results. Consult Sugihara (1986) for additional geometric constraints which prevent interpretation of line drawings which cannot be formed by actually imaging a blocks world object. The paper by Malik (1987) shows how to extend the catalogue of line and junction labels to handle a large class of curved objects. Work by Stockman et al (1990) shows how sparse depth samples can be used to reconstruct and interpret incomplete line drawings of scenes that are much more general than trihedral blocks world. The two-volume set by Haralick and Shapiro contains a large collection of information about the perspective transformation.

In a popular paper, Marr and Poggio (1979) characterized the human stereo system in terms of information processing and proposed an implementation similar to relaxation. The paper by Tanimoto (1998) includes a section on how stereograms can be used to motivate mathematics; in particular, it includes some color stereograms which humans can view to perceive 3D shape. Creation of stereo material for human viewing is an active area of current research; for example, Peleg and Ben-Ezra (1999) have created stereo scenes of historical places using a single moving camera. Automatic focusing mechanisms abound in the commercial camera market: it's clear that depth to scene surfaces can be computed by fast cheap mechanisms. The work of Krotkov (1987), Nayar *et al* (1992), and Subbarao *et al* (1998) provides background in this area.

1. H. Barrow and J. Tenenbaum (1978), *Recovering intrinsic scene characteristics from images*, in **Computer Vision Systems**, A. Hansom and E. Riseman, eds, Academic Press, New York.

2. E. Charniak and D. McDermott (1985), **Artifical Intelligence**, Addison-Wesley.

3. M. Clowes (1971), *On Seeing Things*, Artificial Intelligence, Vol. 2, pp79-116.

4. J.J. Gibson (1950), **The Perception of the Visual World**, Houghton-Mifflin, Boston.

5. R. Haralick and L. Shapiro (1992/3), **Computer and Robot Vision, Volumes I a nd II**, Addison-Wesley

6. D. Huffman (1971), *Impossible Objects as Nonsense Sentences*, in **Machine Intelligence, Vol. 6**, B. Meltzer and D. Michie, Eds., American Elsevier, New York, pp295-323.

7. J. Kender (1980), **Shape from Texture**, Ph.D. dissertation, Dept. of Computer Science, Carnegie Mellon Univ., Pittsburg, Pennsylvania.

8. J. Koenderink (1984) *What does the occluding contour tell us about solid shape?*, Perception, Vol 13.

9. E. Krotkov (1987) *Focusing*, *International Journal of Computer Vision*, Vol. 1 (1987)223-237.

10. J. Malik (1987), *Interpreting line drawings of curved objects*, International Journal of Computer Vision, Vol 1, No. 1.

11. D. Marr and T. Poggio (1979), *A computational theory of human stereo vision*, **Proceedings of the Royal Society**, B 207(1979)207-301.

12. D. Marr (1982), **Vision: a Computational Investigation into the Human Representation and Processing of Visual Information**, W.H Freeman and Co., Newy York.

13. S. Nayar (1992) *Shape from Focus System*, **Proc. Computer Vision and Pattern Recognition**, Champaign, Illinois (June 1992)302-308.

14. M. Subbarao and J-K. Tyan (1998) *Selecting the Optimal Focus Measure for Autofocusing and Depth-from-Focus*, **IEEE-T-PAMI**, Vol. 20, No. 8 (Aug 98)864-870.

15. S. Peleg and M. Ben-Ezra (1999), *Stereo Panorama with a Single Camera*, **Proc. Computer Vision and Pattern Recognition**, Fort Collins, CO, (23-25 June 1999) Vol. I, 395-401.

16. I. Rock (1983), **The Logic of Perception**, A Bradford Book, MIT Press, Cambridge, MA.

17. A. Rosenfeld, R. Hummel and S. Zucker (1976), *Scene Labeling by Relaxation processes*, IEEE Trans. SMC, Vol. 6.

18. G. Stockman, G. Lee and S.W. Chen (1990), *Reconstructing Line Drawings from Wings : the Polygonal Case*, Proc. of Int. Conf. on Computer Vision 3, Osaka, Japan.

19. K. Sugihara (1986), **Machine Interpretation of Line Drawings**, MIT Press, Cambridge, MA.

20. S. Tanimoto (1998), *Connecting Middle School Mathematics to Computer Vision and Pattern Recognition*, **Int. Journal of Pattern Recognition and Artificial Intelligence**, Vol. 12, No. 8 (1998)1053-1070.

21. D. Waltz (1975), *Understanding Line Drawings of Scenes with Shadows*, in **The Psychology of Computer Vision**, P. Winston, Ed., McGraw-Hill Book Co., New York, pp 19-91.

22. P. Winston (1977), **Artificial Intelligence**, Addison-Wesley, Reading, Mass.