Chapter 25

Image Clues to the Scene Puzzle

Local to global transition via simple parametrized models - The hough transform. IMage segmentation revisited. Mumford Shah.

fitting inverse problems together; the web of interacting constraints; Intersection of constraints: shadows and light sources; depth and shadows; distance cues;

Inference questions in vision are unavoidable.

25.1 Introduction

Our studies of vision in the frog and the fly suggested that there were rather straightforward clues in the image to the relevant aspects of structure in the world. Flies, for example, looked to the frog as dark small spots moving at a given velocity; should this spot turn out to be an image of an airplane in the sky then the frog has simply wasted a jump. Such false positives in the relationship between image structure and behaviour must be sufficently rare that, at least for frogs, the energetic cost is tolerable.

Such tradeoffs may well be common in engineered worlds as well. Regardless of the complexity of packaging in the supermarket, the bar code is relatively easy to identify. And, as is clear from examining the returns from a Google query, we're quite tolerant of many false positivies.

In all of these cases, the difficulty arises because the inverse map is many-to-one. We looked at this for two examples of shape-from-shading, the first with sunny-day illumation and the second with cloudy-day illumination. While this discussion opened the way in which local "pieces" interact, the sheer number of assumptions required to make it work seemed daunting. Classical approaches to modeling seek those assumptions – and a framework for using them – that make the problem well-formed and, hopefully, tractable.

Our goal, in this lecture, is to build on the background developed thus far, but to try to "unpack" the inverse map into appropriately-sized chunks, or at least to work out a new approach to solving this problem – a problem we refer to as the scene puzzle. The key difference is that the "assumptions" are not independent: just as values of the gradient at one position constrain the surface at nearby positions, the assumptions made to solve one problem may constrain the formulation of "nearby problems." We illustrate these interactions in the next few sections.

25.2 The Scene Puzzle

Stage designers can be manipulative. To create excitement in a scene, for example, a vantage point can be chosen so that the relevant players are visible to the viewer in a highly controlled manner: such as a zombie approaching a child.

viewer but not to the player Because a scene might be filmed from a single vantage point

At any moment in time only a small portion of our environment is visible

The image as a source of clues for solving the scene puzzle. But the clues are not unique – hence it's a constraint satisfaction problem.

This doesn't imply that the image is a simple mosaic of clues: sometimes a part of the image is; and sometimes not. The former case this is the argument for segmenation (as in the blood cell example). the complexity of edge detection suggests it's a question of organization. Shape from shading tells us that some of the ingredients necessary to solve the organizational puzzle are not even in the image directly (light source).

When we switched to more complex tasks for primates (or robots), for example recognizing a face or a decent lunch in an otherwise (visually) confusing environment, such simple image clues were difficult to use. Even using simple, local "edge" detectors to identify grasping points on automotive parts embedded in a bin of similar parts was frustratingly difficult. Edges are more than just intensity distributions; while it is true that a dark object against a light background will cast a certain bright/dark intensity profile, the physical cause of this profile is not well posed. In order to structure it, we studied the shape-from-shading problem with a single point source, Lambertian reflectance, no mutual illumination, and a half-dozen other constraints. In effect, this amounted to using the $I = \rho N \cdot L$ model as a constraint to pick out which surface normals could work near to one another on the surface, as dictated by the intensity variation.

However, there's much more to image intensity variation than $N \cdot L$; in fact, there are many different physical situations that can give rise to essentially the same local intensity distribution. Focal blur, penumbral blur, and shading can all be shown to give rise to very similar intensity distributions. Examples in which this is approximately sigmoidal—i.e., dark with a smooth transition into white—are shown in Fig. 25.1.

Our studies in the previous two lectures revealed another example of this effect: a given shading distribution could be caused by a distant light source and a simple, Lambertain object (shape-from-shading on a sunny day) or a complex distribution of objects obscuring a distributed light source (shape-from-shading on a cloudy day).

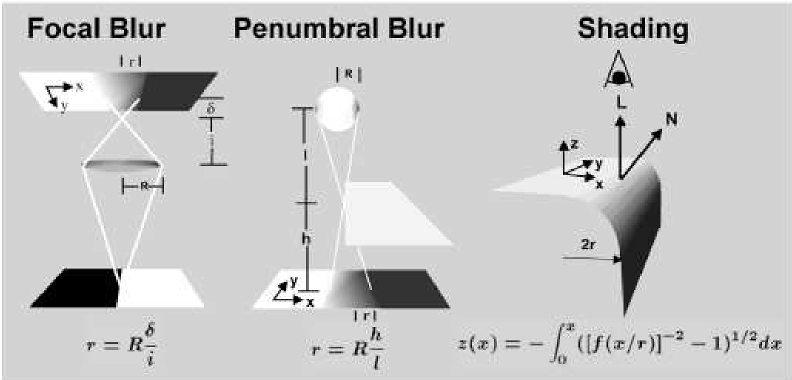

Figure 25.1: Different physical causes can give rise to an intensity distribution that has a sigmoidal structure. Possible physical causes derive from Figure from Elder-Z.

How can these different scene domain models be inferred from the image? How do different reflectance models interact; and how might they be identified? How can material changes be differentiated from shadow edges? And why do images with very different, non-physical optical origins, such as electron microscope images, still make visual sense? These questions derive from what we refer to as the scene puzzle.

In this lecture we take stock of the many different aspects to the scene puzzle that have already been introduced, so that we may develop a richer view of how to interpret image clues that could contribute to solving the puzzle. We can no longer avoid thinking about image inferences. Our current goal is to outline many of the possible constraints that have been identified, with the hope of discovering a way to organize them. Some of these constraints are high-level, abstract scene characteristics that relate to models and some are lower-level confusions that relate to intensity ambiguities.

25.3 Local vs Global Relationships

In the course of our models, we encounted some relationships that are extremely local, such as the intensity given a particular surface normal, or the change in intensity due to a small change in the surface normal. And we encountered some that were global, such as an entire surface being Lambertian.

In effect: local corresponds to "in the small" and global to "in the large."

25.4 Constraints and Consistency

At a highest physical level, models specify an arrangement of objects, materials, light sources and atmospheric effects.

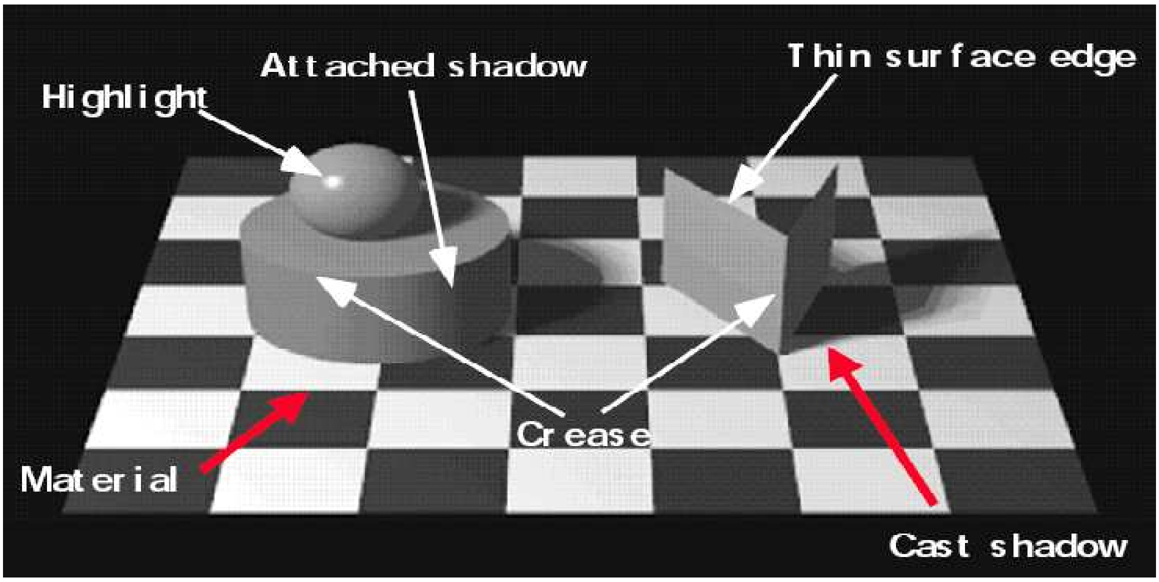

Figure 25.2: At an intermediate-level of description for images, there are several different "semantic" types. Image: D. Kersten

Considering a simple scene that consists of a box and a ball on a checkerboard table top, with a bent card, shows that such a high level of description is perhaps suitable for conjuring objects in memory, but is not really very visual. Little information about the arrangement geometry is given, and nothing about the material properties. Providing more visual clues, such as highlights and shadows, paints and creases, gives a more vivid impression; see Fig. 25.2.

With these primitives in place, we can now begin to reason about whether this set of objects makes sense: do the highlights suggest light source positions that could be responsible for the shadows? Are the necessary obscurations in place? Some of this reasoning resembles the physical analysis just completed, while other aspects of it suggest a more qualitative group of questions: even though we lack a detailed physical model, do the components make sense together: *are they consistent* according to the general properties of our visual world? Answering such questions is like solving a puzzle, but a special kind of puzzle with multiple ambiguities. This is precisely what Helmholtz called UNCONSCIOUS INFERENCE.

> ...The general rule determining the ideas of vision that are formed whenever an imporession is made on the eye ... is that *such objects are always imagined as being present in the field of vision as would have to be there in order to produce the same impression on the nervous mechanism ...*
>
> ...The psychic activities that lead us to infer that there in front of us at a certain place there is a certain object of a certain character, are generally

> not conscious activities, but unconscuius ones. In their result they are equivalent to a *conclusion*, to the extent that the observed action on our senses enables us to form an idea as to the possible cause of this action. ... But what seems to differentiiate them from a conclusion, in the ordinary sense of that word, is that a conclusion is an act of conscious thought. An astronomer, for examples, comes to real conscuius conclusion of theis sort, when he coputes the positions of the stars in space, their distances, etc., from the perspective images he has had of them at various times...His conclusion are based on a conscious knowledge of the laws of optics. In the ordinary acts of vision this knowledge of optics is lacking. Still it may be permissible to speak of the psychic actic of ordinary perception as *unconscious conclusions* ... Helmholtz, Treatise on Physiological Optics, vol. III, pp 2 - 4.

25.4.1 Toward A Logic of Vision

It is tempting to follow Helmholtz strictly, and to imagine that each of the notions above, highlight, shadow, object, etc, have the logical status of a proposition. Then we can reason with them in a formal manner.

Consequences of such reasoning:

- Consistency now has a formal meaning (modus ponens, etc).
- Adding a new "fact" to a database of facts can completely change the character of a conclusion from the database. Example: simply adding a small change in an image can lead to a large change in its interpretation; local constraints can have global effects. Kersten shadow examples: see Fig. 25.3 (top).
- It is not even necessary to have some confirming evidence, provided there is no inconsistent evidence. Thus one of the most important aspects of such inferences is creating structure. See Fig. 25.3(bottom)
- Multistable percepts can arise when a local interpretation changes; again, this can have global effects. See Fig. 25.4.
- In a system composed of interacting hypotheses, it is necessary that additional structure can be added; this is a form of composition of hypotheses; See Fig.fig:kersten-card.
- Finally, we note that the logic of it all can sometimes override the actual physical data; this is another example of "what you see ain't necessarily what's out there." (Mach) Example from Adelson in Fig. 25.6

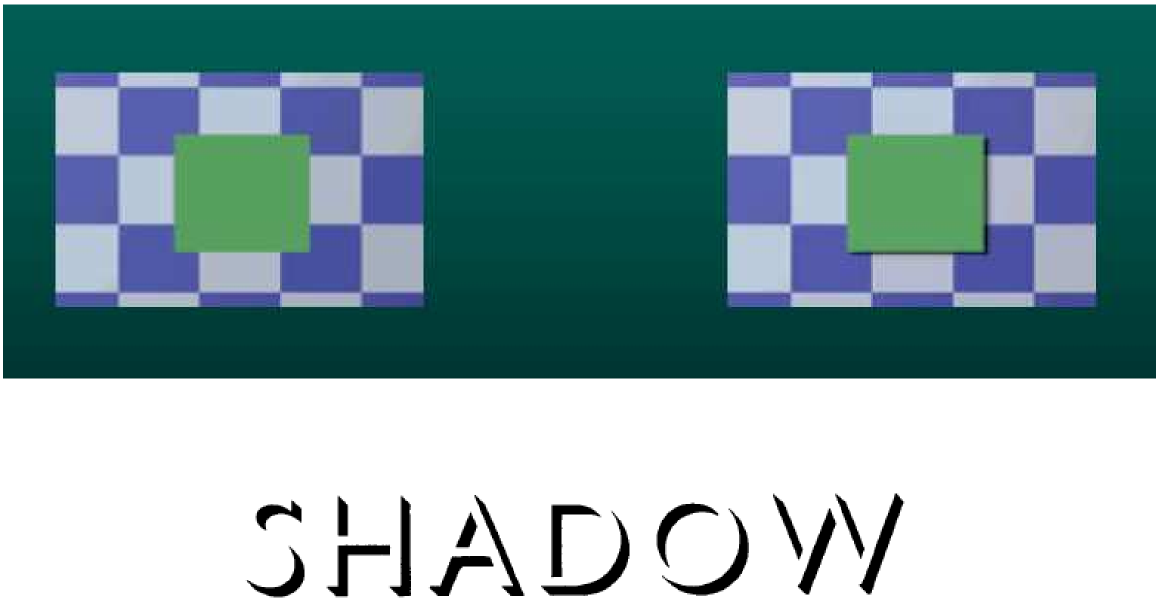

Figure 25.3: Shadows can imply depth. (top) Adding a small shadow on the right side of the square completely changes the global construct of the square. Image: D. Kersten. http://www.graphics.cornell.edu/workshop/talks/perception/kersten.pdf (bottom) It is not even necessary to have a visible object that casts the shadow (figure: R. Gregory, Eye and Brain).

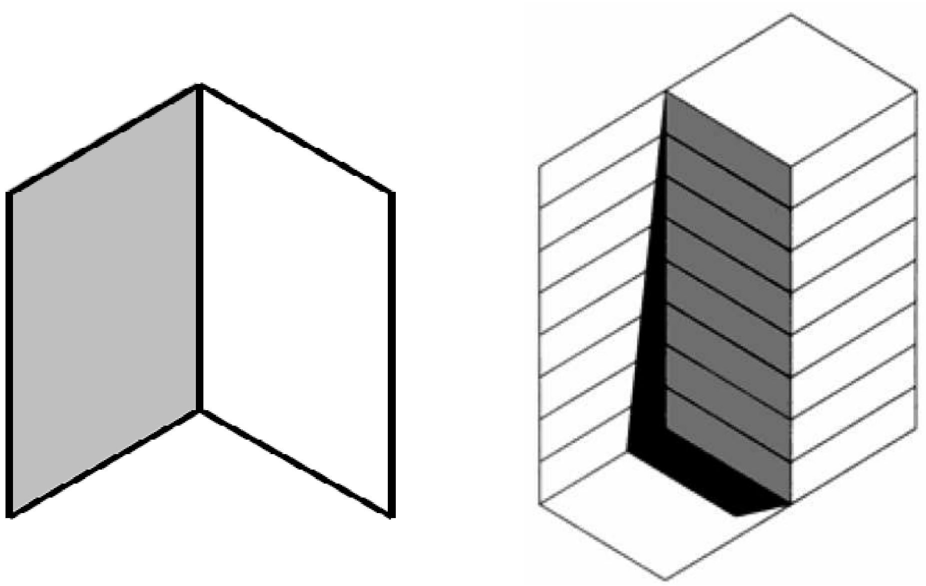

Figure 25.4: (top) Mach card illusion illustrates ambiguity of image interpretation. Note how outline can shift from a card bent convexly to a card bent concavely. Roof vs valley. Light source shifts as well. (bottom) Mach's corner illusion (from Gregory's webpage). Notice how apparent brightness seems to change with multistability of the image. Notice also how the apparent light source position changes as well. Also do Langer's example of the face/vase.

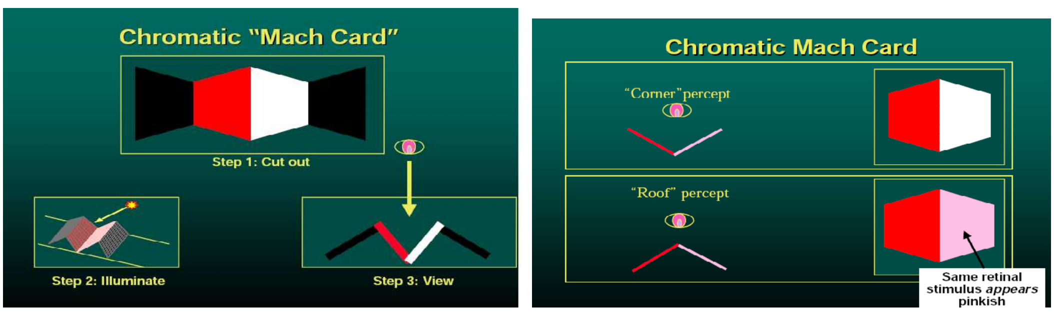

Figure 25.5: The perceived surface dictates the perceived color. Extension of the Mach folded card illusion by Kersten to include mutual illumination effects. Image: D. Kersten. The Mach card should be made physically to see it work; A. Hurlbert, Perception 1998, volume 27, remarks how it is difficult (if not impossible) to make this work on a self-luminant display.

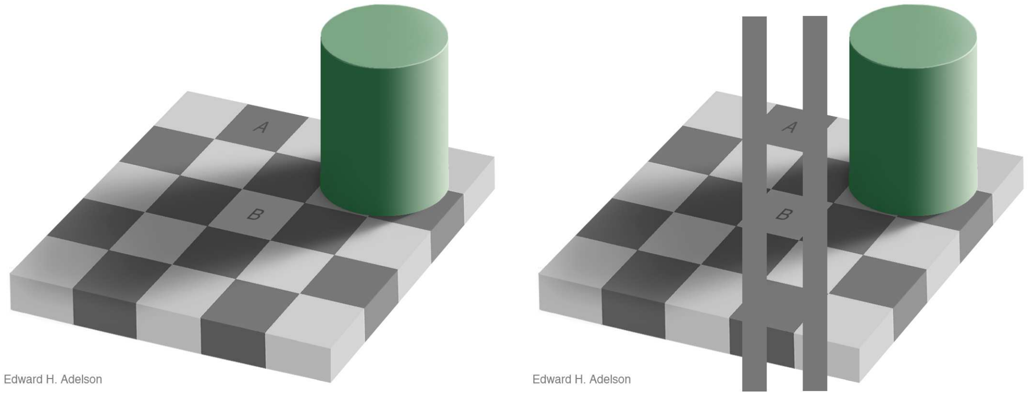

Figure 25.6: Adelson checkerboard. Appearance is a function of interpretation; and only indirectly connected to the photoreceptor responses.

25.4.2 ...and a Network of Visual Areas

It is tempting to return to the Felleman/van Essen illustrations of visual cortex, and to the Zeki functional interpretation that each visual area supports a distinct function. If V1 is doing edge detection, V2 stereo, V4 color, IT object recogntion, etc, then the feedforward and the feedback connections provide the constraints between the relevant propositions. See Fig. 25.7.

An advantage of thinking in these terms: all of the brain is explained. And we know how many, and which, visual functions to be seeking. But there are problems.

25.4.3 Connectivity between Visual Areas

Revisit the van essen hierarchy notion, and relate this back to Hubel and Wiesel. The connectivity between areas is a graph. This would be OK at the function level, except a more detailed view also suggests a graph structure. This question needs to be elaborated upon.

(see Vezoli et al. for figures)

25.4.4 Conspiracy vs General Position

While high-level reasoning about the scene is an attractive possibility, and certainly provides part of the answer, by itself it is not sufficent. Sometimes the image is made from an unfamiliar viewpoint, like this aerial example in which the picture is taken from the top; the shadows are not the animals! See Fig. 25.8.

A more complicated issue arises from conspiracy. With serious effort one can compose a collection of material that is not in arbitrary position to give the impression via shadowing of something that is not there. Such configurations are fragile, in the

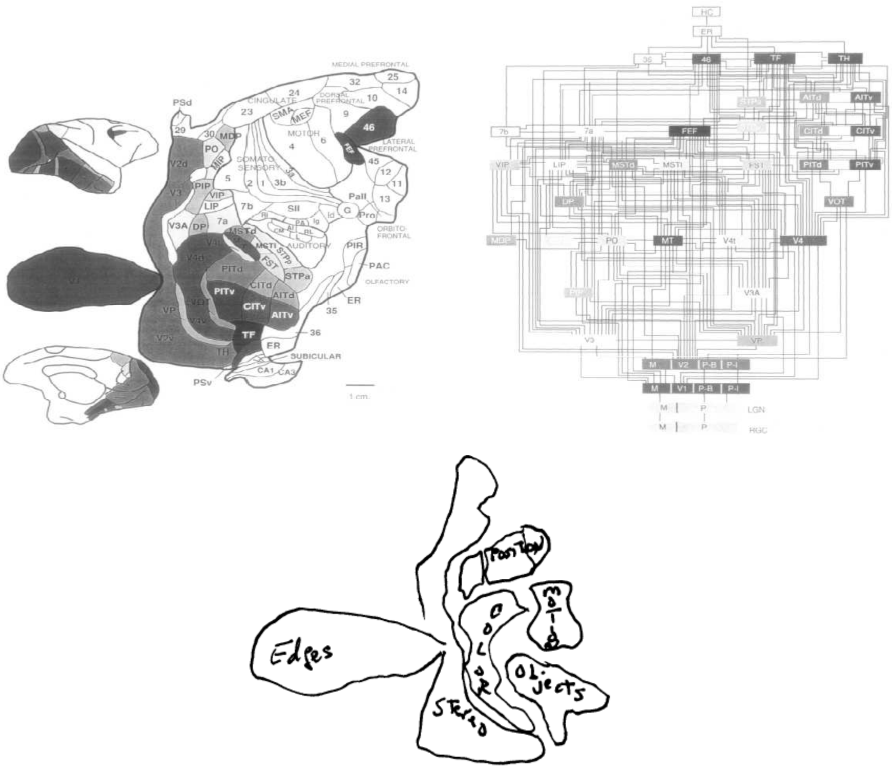

Figure 25.7: Diagram of visual areas. According to Zeki, each area has developed to implement a particular visual function. Some of these different functions are written into the lower tracing. (Notice how similar this is to the popular view of cortical processing from fMRI!) From:Felleman DJ, Van Essen DC. Distributed hierarchical processing in the primate cerebral cortex. Cereb Cortex. 1991 Jan-Feb;1(1):1-47.

Figure 25.8: An aerial image of camels in the desert. Because the sun is very low in the sky, the shadows are long and our temptation is to see them as the animals. This is incorrect. http://www.moillusions.com/2006/09/national-geographics-shadow-camels.html

Figure 25.9: Shadow illusions constructed by arranging arbitrary objects in a manner that gives rise to a coherent shadow impression. (left) www.moillusions.com/2008/01/block-shadow-girl-optical-illusion.html Note: a small change in any of the blocks could destroy the effect. (right) www.moillusions.com/2007/06/amazing-shadow-illusions-collection.html shadow illusion created by Tim Noble and Sue Webster.

sense that a tiny movement of any component could destroy the whole illustion. See Fig. 25.9.

We assume a normal view of the world –what does this imply? What are the proper expectations from it? That familiar objects are illuminated from familiar viewpoints in familiar configurations? By putting probabilitity distributions onto configurations one could gather information about this – but how are such distributions learned? represented? Clearly, in a normal view

25.4.5 Sources of Ambiguity

The analysis of constraints need not be high-level, nor need it be completely logical.

intersection of constraints in motion

25.5 The Scene to Image Map

We saw in the previous examples that many different scenes map to the same image, which makes the inverse problem not well formulated. But how shall we formulate this map? Recalling the earlier discussion of hierarchies of features extracted from the image, we can now reapproach it from our (now) more informed perspective. In particular, we now know that the image formation map is not, in general, invertible without additional constraint; what should these be? The analysis of shape-from-shading on a sunny day mixed many different kinds of constraint together, from quite general ones (surfaces tend to be smooth) to more specific ones (light source is very small and very far away). Somehow the constraints that are applied have to be appropriate to the level of representation being computed.

25.5.1 General Purpose Models

Not all images are created from scene configurations that are easily classified as sunny-day or cloud-day; See Fig. 25.10. It follows from the hierarchy notion that these should be very general purpose models, models that are applicable to wide classes of scenes even when there is little or no *a priori* knowledge about the class of scenes or how they were generated.

What are the properties of images that will hold across many different classes of scene, whether these are typical of the natural world or of rather different ones? An examination of the EM image suggests that many of the early features we have been talking about make some sense, although we haven't yet developed them in a satisfying manner. So far they have been tied too strongly to specific tasks.

Since they are closer to the image than to scene models, these early, general purpose features should be tied in some manner to image statistics: the image intensity distributions should almost always organize into such features (or into noise, the absence of such features). But what order statistic should be employed? Second-order, neighboring interactions are probably too low, because we needed many more pixels even to define a simple derivative operator that worked. The problem with higher-order statistics is that there are simply too many of them; how can we make a choice?

Another important hint about general purpose features and the general purpose models that they obey is how they organize into additional structure. Some great insights here were obtained by the Gestalt psychologists, some of which we have already seen. See Fig. 25.11. We are able to sense contours along which "one can walk" and surfaces over which "one can slide." Occlusion emerges. But some configurations are just unexpected and difficult to see; See Fig. 25.12. Clearly, once again, the notion of generalized position comes to the fore, because any slight rearrangement of the two tear-shaped circles would destroy the effect. Put another way, this states that small changes in the image should only lead to small changes in the percept (which seems intuitively different from what happened with the shadow example).

Finally, one gets a sense that, since we are able to make drawings of images like the EM example above, whether we have seen it before or not, and that these drawings convey some of the *structure* of the image, somehow this is the right direction for the early stages of the representation.

25.5.2 Constraints, Compositions and Cortical Architecture

The second major issue we shall have to face is what kind of models can we use to put these basic pieces together. Should they be an "erector set" of puzzle parts (how will this support generalization?) or should they be pieces of a grammar (noun phrases vs verb phrases at the general level; words at the specific levels?)? What are our other possibilities?

In addition to these general concerns about the form of the models, we shall also have to worry about how they could be reduced to neurobiological terms? At the high levels should they consist of interactions between visual areas while at the low levels interactions between neurons? Are not all interactions, in the end, between neurons? And why are neurons arranged structurally in columns? Is this an artifact of development or is there an information processing view of it all?

25.5.3 Anatomical and Physiological Diversity

different cell types: pyramidal; basket, etc. Smooth vs spiny.

how do we deal with all of this complexity: Computational anatomy (braitenberg and descendents).

See Fig. 25.14.

25.6 Labeling Graphs

In order to develop the inference structure for doing a formal analysis of vision problems, we need a general model for all types of abstraction. Earlier we saw how trees were the underlying abstraction for studying hierarchies, and how the structure of trees provided real insight into understanding the log-n complexity given an organized search.

For vision we shall require a representation that is more general: GRAPHS. This is a structure that contains a set of NODES, which might represent anything from low-level pixel locations (see Fig. 25.15) to high-level objects, plus a neighbor relationship – EDGES – that connects different nodes together.

Formally, a graph G = ({nodes}, {edges}), where {edges} $\subseteq$ {nodes} $\times$ {nodes}.

Labels can be assigned (mapped) to either nodes or edges in a graph.

and a model for computation (what *does* this word mean?) on the graphs: turing machines as an example. Note: reading the sequential tape seems totally

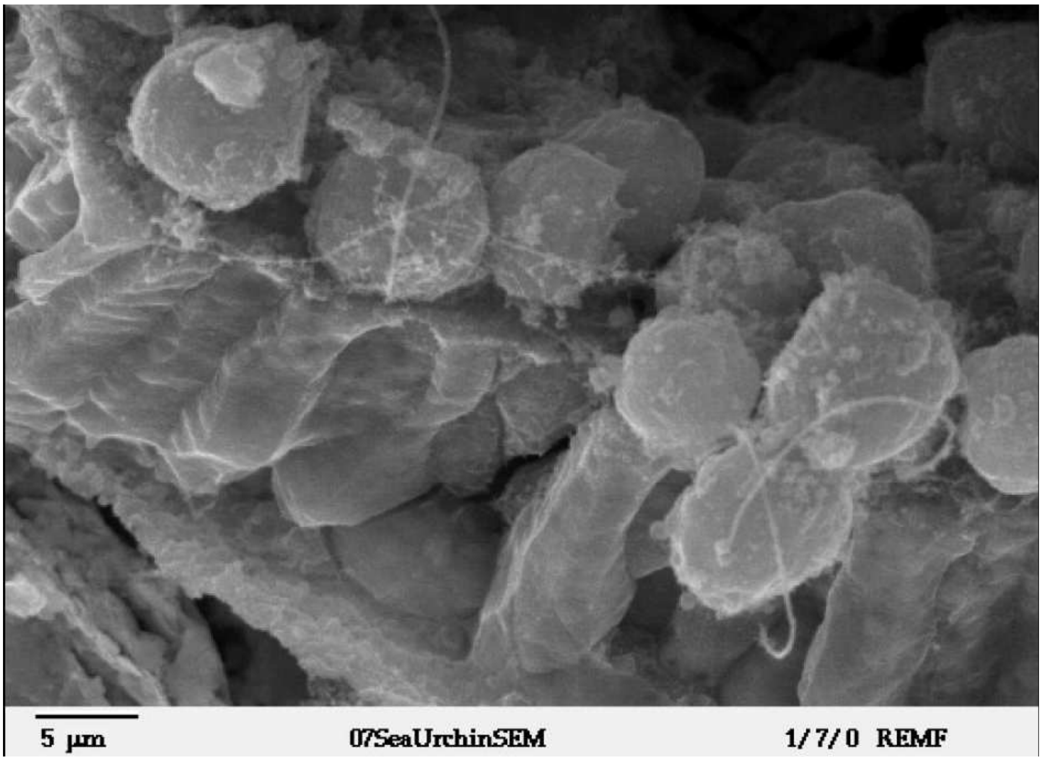

Figure 25.10: Our early visual system supports representations and inferences of structure from images arising from unusual sources. This is an example of a scanning electron microscope image of Strongylocentrotus drobachiensus embryo at the primary mesenchyme blastula stage. Biologists care about the following: Embryo was split open to reveal the outer epithelial layer and the blastocoel cavity at the vegetal pole. At this stage of embryonic development, primary mesenchyme cells are forming(flask shaped cells in the epithelial layer) and beginning to migrate into the blastocoel cavity. In this class we care about the image formation process and what you see. Notice the strong 3-D impression, even though the lighting model is completely different from normal (visible-light) ones. Where, for example is the light source? remf.dartmouth.edu/images/SeaUrchinSEM/source/1.html

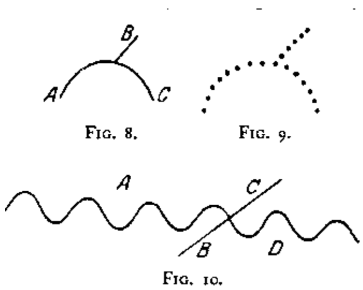

Figure 25.11: Imagine wandering along a curve such as those on the left side, even when it is a dotted curve. One tends to continue in the same direction. Problems can arise in choosing direction at crossings, because then there is a question of which branch to follow. Locally this can be quite confusing (right). Original drawings from Wertheimer 1923.

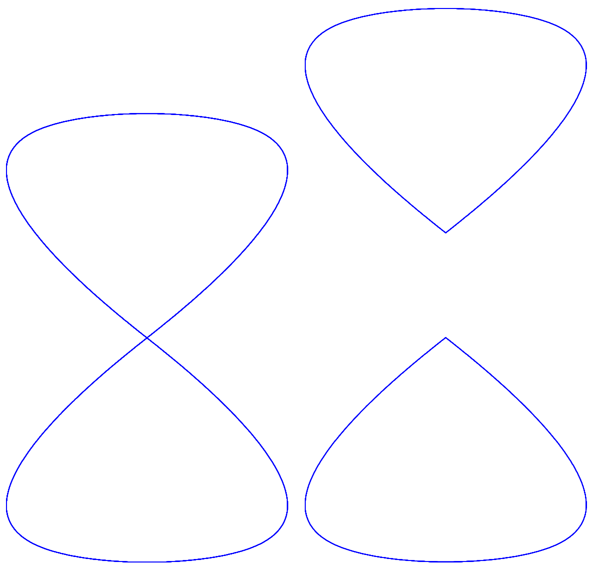

Figure 25.12: (left) The *figure-8* is a classical non-simple curve, which is rarely seen (middle) as two irregular curves meeting precisely at a point. Why? What kind of model would imply this?

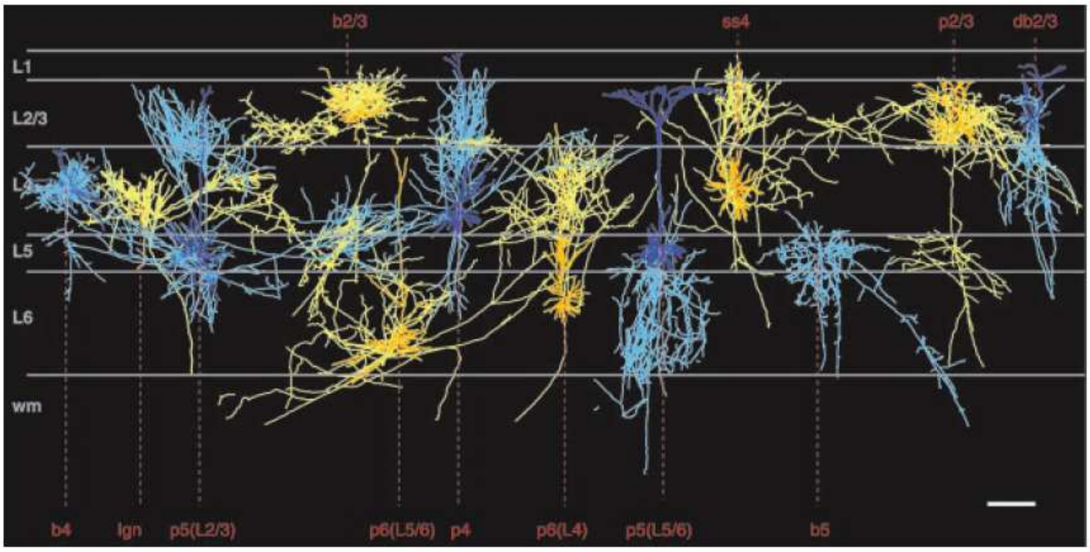

Figure 25.13: Illustration of the different cell types in cat visual cortex. Blue and yellow were used only to aid viewing; the axons for each type are shown in bright blue or bright yellow. Boutons are not shown. Cell types: basket cells (b2/3, b4, b5 in layer 2/3, 4, 5); db: double bouquet; p pyramidal; ss spiny stellate. Coronal view. From Binzinger, Douglas, and Martin.

unlike the “parallel” nature of neural computation. Does this preclude extending results from Turing complexity analysis to parallel systems?

25.6.1 Graph Coloring

One of the easiest constraint satisfaction problems that can be described is that of coloring in the countries on a map so that no two adjacent countries have the same color, which makes it easy to notice the borders between countries. How hard is this problem, and how might it be solved?

The first step is to abstract the map onto a graph, by naturally specifying a node for each country and an edge between those countries that share a border (Fig. 25.16).

Now, we need to assign colors to nodes. Letting $u, v \in \{\text{countries}\}$, colors are assigned by the function $f : \{\text{countries}\} \rightarrow \{1, 2, ..., k\}$ where $f(u) \neq f(v)$ when u is a neighbor of v. Traditionally, we also seek to find the smallest number k that will work for a particular map. (Otherwise it’s trivial: just number the countries with different colors.) Such a function is called a COLORING of a graph G.

Let’s begin by considering $k = 2$. This asks whether there is an assignment of red and green to countries so that all of a red country’s neighbors are colored blue and all of a blue country’s neighbors are colored red. Thinking abstractly, this is a question about the existence of cycles of even/odd length, and its solution boils down

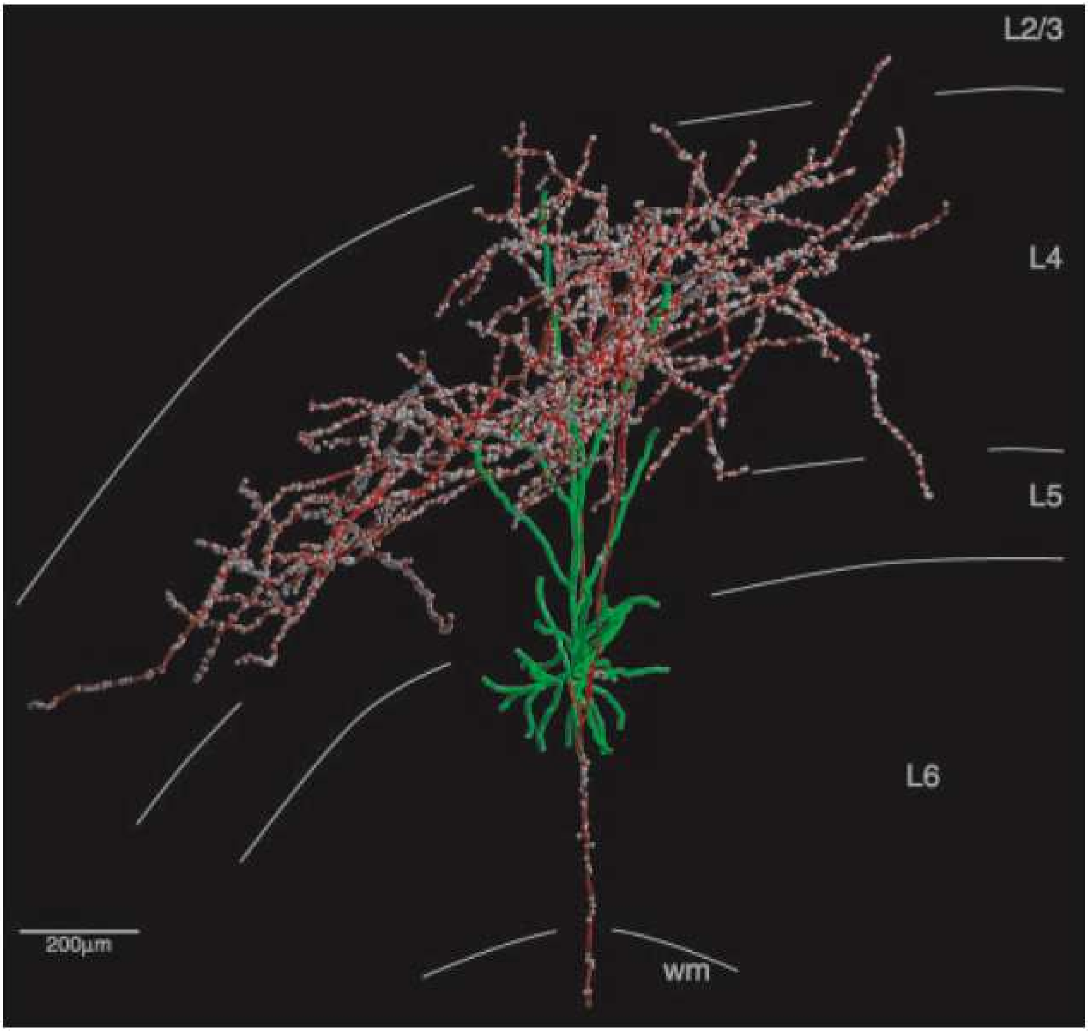

Figure 25.14: Reconstructed layer 6 pyramidal cell. Axon shown in red, boutons in white, dendrites in green. Cortical layer separations as gray curves. Simple receptive field, monocular, preferred orientation 60 degrees, size: 0.3 x 0.5 degrees. Coronal view. From Binzinger, Douglas, and Martin.

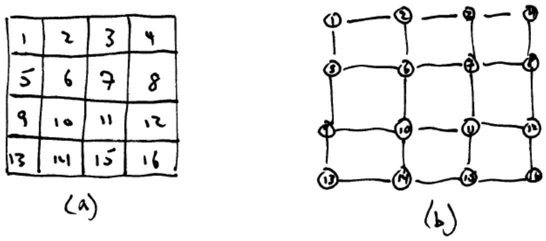

Figure 25.15: Taking an image to a graph abstraction. Note in this case that pixels are labeled by their number, and edges designate which pixels are "neighbors" (in this case, just their north-south and east-west neighbors are defined. Can you think of other neighborhood relationships that might be useful?

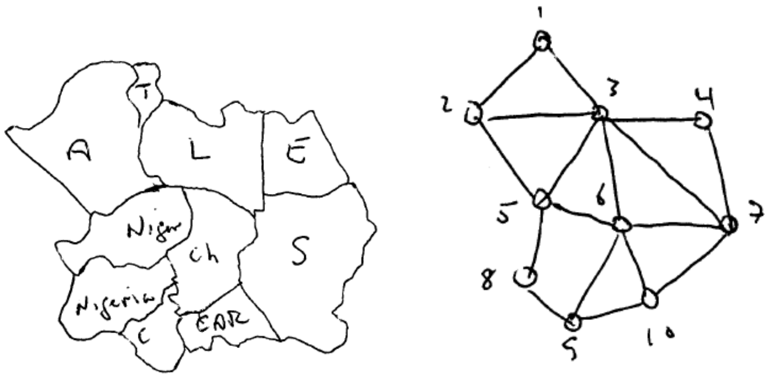

Figure 25.16: Fragment of the map of Africa and its graph representation.

to whether there is a bipartite representation of the graph (Fig. 25.17). In complexity terms this is easy, and can be solved in $O(m+n)$ steps, where m is the number of countries and n is the number of edges.

As soon as $k = 3$, however, things change drastically, and the problem becomes extremely difficult computationally (NC-complete). There are many excellent discussions of this question in the theoretical CS literature - see, e.g., Kleinberg and Tardos, sec 8.7.

bipartite graphs

25.6.2 Line Labeling

Now let's consider a vision problem, perhaps slightly artificial but still informative. Historically this problem arose from the realization that (i) early vision is difficult without some notion of the scene from which it came; (ii) the simplest possible scene domain consists of polyhedral objects viewed from general position with junctions

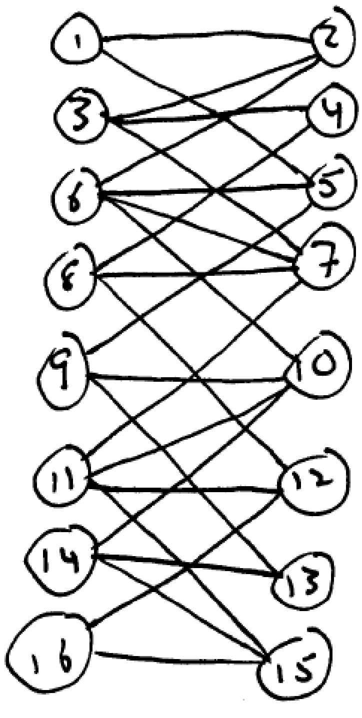

Figure 25.17: 2 coloring is equivalent to finding a bipartite representation.

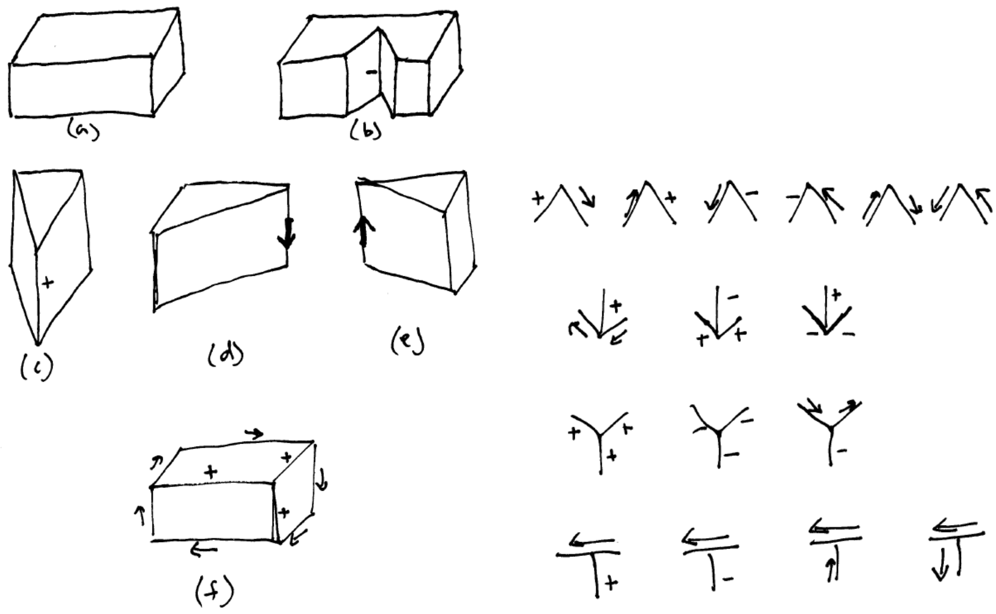

Figure 25.18: The blocks world problem. (left) Examples of different “scenes” from the blocks world. (a) A typical line drawing. The task is to infer what the physical characteristics are for each line. Are they occluding with the object on the left (d) or right (e); are they a juction of two faces meeting convexly (c) or concavely (b)? (right) Not all pairs or triples of labels meeting at a junction are physically possible; the physically correct possibilities are shown here.

formed by up to three faces, or surfaces, with no cracks or shadows. Objects are opaque.

4 line labels implies 4^2 possible ways of labeling an L-junction; and 4^3 ways of labeling “W”, “Y”, or “T” junctions, or 208 different ways in which line labels can meet at the polyhedral vertices. Analysis of the domain indicates only 16 junction types. See Fig. 25.18. The line labeling problem is to take these legitimate junction labeling as constraints to determine whether a line labeling is physically possible (“consistent”) or not.

We shall discuss how to solve this problem in the next lecture. This will require developing the graph abstraction and a method for considering which labels are consistent with each other across vertices.

Note: cycle of consistently-oriented arrows in Fig. 25.18.

25.7 Vision as Inference; neural nets and constraint satisfaction

include the statistical physics comments here

25.8 Conclusions

The jump from image intensities directly to scene structure is too large, in the sense that too many intermediate assumptions have to be made. Verifying these assumptions is as hard as the vision problem, since many are global. (Where are the light sources? What are the surface material properties? etc)

Although the transition is difficult to formulate in a one-to-one fashion, there is information locally. We have to determine what are the components and how they might fit together? In a sense putting this information together is a constraint satisfaction problem, to which we turn next, after which we shall return to the question of what are the primitives.

Chapter 26

Constraint Satisfaction in Columns

Discrete Constraint Satisfaction Problem. Discrete Relaxation Labeling. Relaxation Labeling: nodes, labels, compatibilities, Label Discarding Rule, Average Local Consistency.

Relaxation Labeling

26.1 Introduction

Different features (colors; edges) are represented at each point in the image: how do they fit together? One question is how do they fit together across visual space? For example, how do nearby image intensities or nearby edge elements interact? How should they interact? At a different level we must address how the different features fit together? For example, how do edges and colors interact?

Development needs work: the idea is to use constraints "in the large" in the previous chapter and "constraints in the small" here to contrast against one another. The Waltz labeling stuff can go into the other chapter.

26.1.1 Parallel Iterative Algorithms

To set the stage for developing constraint satisfaction—and other—algorithms that could be implemented in a simple abstraction of neural architecture, we need to recall certain aspects of the Limulus lateral plexus. It consisted of a

· set of input units (the photoreceptors);

· connections between input units (lateral axonal interactions);

· local processor (the neurons in the plexus);

· set of outputs (the axons from the local processors).

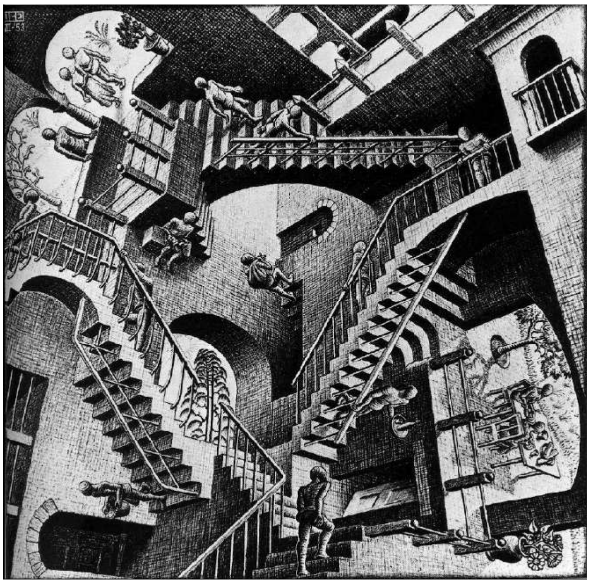

Figure 26.1: Escher

In the feedforward model of the plexus one could imagine multiple copies of the processor units, with the output of one set to be the input to the next.

In the recurrent model of the plexus there was an iteraction, in which the output of the local processor units was fed back as an input until a steady-state, or equilibrium, was reached. It is this steady state that is the relevant output.

Basic abstraction:

· initial values

· iteration

· convergence

Now, to give these notions some flesh we shall do several examples. The first continue our discussion of differential equations, the second is a discrete, high-level example from scene labeling, and finally, in the next lecture, a low-level example from "edge detection." This will set the stage for a deeper realization of how contours could be inferred from images.

26.2 Solution of Laplace's Equation

In our discussion of the heat equation (Sec. ??), we described how an initial distribution of heat diffused to an equilibrium subject to boundary conditions. Remember, the heat equation $u_t = \Delta u = \frac{\partial^2 u}{\partial^2 x} = 0$ at equilibrium, which states that Laplace's equation governs this equilibrium.

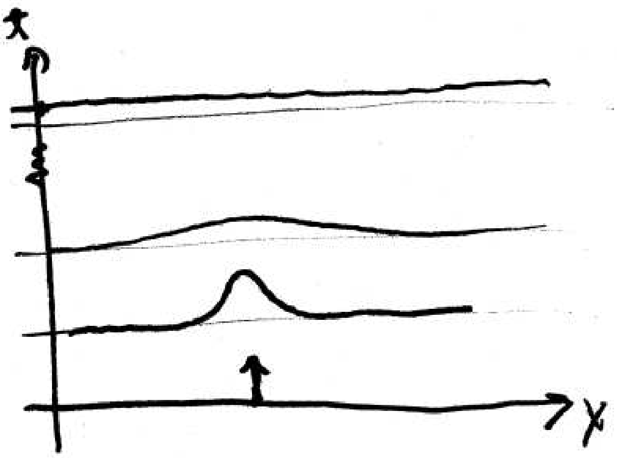

Figure 26.2: 1-D solution of heat equation

Problem: find $u(x)$ such that it satisfies:

$$\Delta u(x) = \frac{\partial^2 u}{\partial x^2} = 0$$

subject to the boundary conditions:

$$u(1) = 0 \qquad \text{and} \qquad u(6) = 5.$$

26.2.1 Southwell Relaxation Solution

We have already seen how to discretize differentials for functions in the Chapter on edge detection. We now apply this to the discretization of the heat equation and obtain:

$$\frac{u(i,n+1) - u(i,n)}{\Delta t} = \frac{u(i-1,n) - 2u(i,n) + u(i+1,n)}{(\Delta x)^2} \tag{26.1}$$

where the notation is explained in Fig. 26.3. Now, if we choose a discretization such that

$$\lambda = \frac{\Delta t}{(\Delta x)^2}$$

then

$$u(i,n+1) = \lambda u(i-1,n) + (1-2\lambda)u(i,n) + \lambda u(i+1,n)$$

which takes on a straight average when $\lambda = 1/3$ is chosen:

$$u(i,n+1) = 1/3(u(i-1,n) + u(i,n) + u(i+1,n)).$$

This formula is Southwell's relaxation.

An important property of it is seen when random INITIAL DATA are chosen for prespecified BOUNDARY CONDITIONS.

26.2.2 Discrete Relaxation Labeling Solution

(use this as an introduction to rlp?)

There is a drastically different way to think about solutions to Laplace's equation, which involves explicitly enumerating all of the possibilities, and then checking them for consistency.

Recall from the section on edge detection, that we can discretize the Laplacian operator (in one dimension) to obtain:

$$\Delta = \boxed{1 \mid -2 \mid 1}$$

The enumeration can be done in tabular form, which, in one dimension, looks as follows:

	5	5	5	5	u=5
	4	4	4	4	
	3	3	3	3	
	2	2	2	2	
	1	1	1	1	
u=0	0	0	0	0	
x=1	2	3	4	5	x=6

Now, observe that for one of the values of the solution function $u(x)$ to be possible at a position x, it is necessary that there is a possible value for $u(x-1)$ and a possible value for $u(x+1)$ such that

$$C_{x-1,x,x+1} = u(x-1) - 2u(x) + u(x+1) = 0.$$

This amounts to taking the Laplacian operator, viewing it as a template, evaluating it over the table (image), and checking to see where it evaluates to 0. That is the key constraint connecting possible values of u at adjacent positions. Any values of u that do not satisfy it can not be part of the solution; i.e., are impossible.

To illustrate, consider the value $u(3) = 3$. Suppose we choose $u(2) = 2$ as one possibility; then we have $u(4) = 2 \times u(3) - u(2) = 2 \times 3 - 2 = 4$, one of the possibilities for $u(4)$. Thus, for this situation, the Laplacian constraint does not actually constrain. However, the situation is drastically different when one considers points nearer the boundaries. With $u(1) = 0$ fixed, given the range of possibilities at $u(3)$ we see that the only values at $u(2)$ satisfying the Laplacian constraint are $u(2) \in \{0, 1, 2\}$. Similaryly, at the other boundary, we obtain $u(5) \in \{3, 4, 5\}$.

The above computation can be carried out for each possible value of $u(x)$; in fact, if we are careful to differentiate between the intermediate values being computed, and

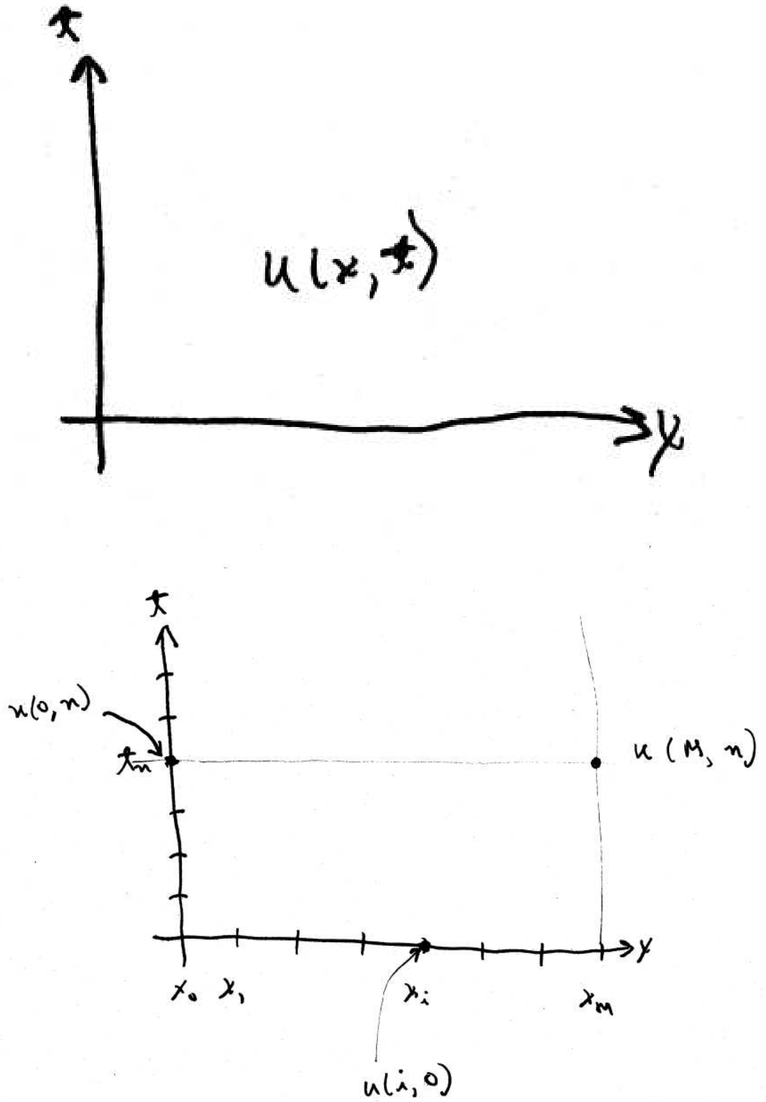

Figure 26.3: Discretization of the variables for solving the heat equation in 1 spatial dimension x.

those stored in each location in the array, then each one can be checked in parallel. The basic algorithm is:

```
{\
For each possible value of $u(x)$ at each position $x = 2,3,4,5$\\
  check to see if there exist values of $u(x-1)$ and $u(x+1)$ such that\\
    C_{x-1, x, x+1} = 0;\\
  If not, then mark that value for $u(x)$ as impossible\\
}
```

Deleting impossible values from the chart, we have:

		5	5	5	u=5
		4	4	4	
		3	3	3	
	2	2	2		
	1	1	1		
u=0	0	0	0		
x=1	2	3	4	5	x=6

Now, we immediately notice that the boundary values have begun to propagate their influence inward, and that certain of the possible values have been eliminated from the positions adjacent to the boundaries. However, some of these values may have been necessary to support other values, so now we had better re-apply the constraint application rule to determine whether other values should be eliminated:

			5	5	u=5
			4	4	
		3	3	3	
	2	2	2		
	1	1			
u=0	0	0			
x=1	2	3	4	5	x=6

Continuing...

				5	u=5
			4	4	
		3	3		
		2	2		
	1	1			
u=0	0				
x=1	2	3	4	5	x=6

					u=5
			4	4	
			3		
		2			
	1	1			
u=0					
x=1	2	3	4	5	x=6

					u=5
			4	4	
			3		
		2			
	1	1			
u=0					
x=1	2	3	4	5	x=6

					u=5
				4	
			3		
		2			
	1				
u=0					
x=1	2	3	4	5	x=6

					u=5
				4	
			3		
		2			
	1				
u=0					
x=1	2	3	4	5	x=6

Each of these tables represents an iteration, so this algorithm took 6 iterations before convergence. *Convergence* occurs when the table is unchanged under the value discarding rule.

convergence

iteration

explicit vs. implicit representation

26.3 Southwell Relaxation and Iterative Algorithms

Our previous explorations into physical models for vision problems raised

this is a way to take random, noisy estimates

26.3.1 Harmonic Functions

Something really interesting has taken place in our use of the laplacian. Remember, this is an operator that says how much the "center" value differs from its "surrounding" values.

In our first encounter, in the retina, it lead to Mach bands: amplified differences. This was then exploited as an edge detector, or the attempt to find those locations in the image where intensities differed a lot.

However, in the differential equation setting diffusion arose – this is more of an attempt to get at smoothness issues. Perhaps the best expression of this is in the discrete relaxation.

Definition of harmonic function. In 1-d, this are the straight lines. (two dimensional space of functions). In d¿1 dimensions, spaces of harmonic functions are infinite-dimensional. See discussion in Lawler, Random Walk and the Heat Equation.

26.4 Discrete Constraint Satisfaction

(from Rosenfeld volume)

Constraint satisfaction techniques were introduced to reduce search complexity by pruning branches that could not be part of global solutions. It was in this context that Waltz realized that the set of labels gave rise to a dictionary of constraints–only certain types of labelled lines could meet at a junction (see Fig. 26.5 caption)–and that this constraint could be exploited to locally prune many branches of the search tree before the global search. Such *Waltz filtering* was serial: a process "wandered" from vertex to vertex in the line drawing, and deleted those labels that couldn't possibly arise from a line drawing of a scene in the blocks world.

Rosenfeld motivated by his research in automata theory [**?**] and it had the important effect of placing parallel constraint satisfaction techniques on a sound mathematical footing; for related and subsequent work see e.g., [**?**, **?**, **?**].

The original formulations of relaxation labeling were as follows, building from the discrete to the continuous.

We begin with the abstract structure of discrete relaxation labeling, the formalization of Waltz filtering and, more generally, constraint satisfaction, by parallel label discarding [**?**, **?**]. The basic idea is to take a collection of objects (in the Waltz case, lines) and define a neighbor relation over them (lines that meet at junctions are neighbors). Attached to each line is a set of possible interpretative labels, and blocks-world semantics specify which labels can meet at a junction. Those labels that form impossible pairs (i.e., that have no possible continuation along each of the lines meeting it at a junction) are discarded. Unlike Penrose's impossible triangle and the bizarre worlds of M. C. Escher, these configurations cannot arise in our physical existence.

Formally, let $i \in I$ denote a discrete object in the set I, and let $\lambda \in \Lambda_i$ denote the set of labels attached to object i. Let $neigh(i)$ denote the neighbor relation over

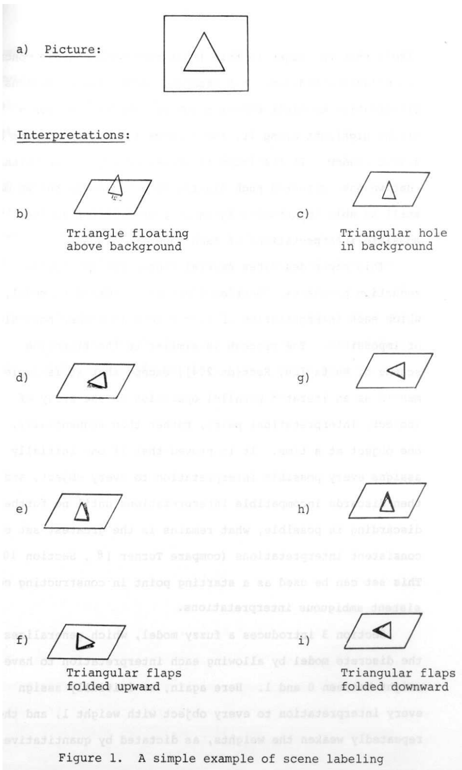

Figure 26.4: The "triangle world" semantics for line labeling, following after an example in the original paper[?]. This is about the simplest possible labeling world, a layered 2-D universe. Only three line labels arise: $+$ denoting a convex edge between two planar surfaces; $-$ denoting a concave edge; $\rightarrow$ denoting a occluding edge (with the occluder to the right while facing in the direction of the arrow). The constraint dictionary is equally simple, and consists of $(\rightarrow, \rightarrow) \in \Lambda_{i,j}(\lambda, \lambda')$, describing a junction floating above the background; $(\leftarrow, \leftarrow)$, denoting a junction "hole" below the background; $(\rightarrow, -)$ and $(-, \rightarrow)$ a concave edge meeting an occluding edge; and $(+, \leftarrow)$ and $(\leftarrow, +)$ denoting a convex edge meeting an occluded edge.

CHAPTER 26. CONSTRAINT SATISFACTION

objects, with $j \in neigh(i)$ a specific neighbor of i. We therefore have a graph with sets of labels attached to each node.

Now we introduce constraint. Let the legal label pairs be denoted $\Lambda_{i,j}(\lambda, \lambda') \subseteq \Lambda_i \times \Lambda_j$, which should be read as label λ' at node j is consistent with λ at i. In the triangle-world, example legal pairs are listed in the caption to Fig. 1.

A label λ at i is impossible, and should be discarded, if there does not exist at least one label $\lambda' \in \Lambda_j \quad \forall j \in neigh(i)$ such that $\Lambda_{i,j}(\lambda, \lambda')$. Notice that iteration becomes necessary, because the particular label λ' at j that was supporting a label λ at i may have been discarded, so there is no further context to support λ at i. Observe, moreover, the inherently parallel aspect to this statement, because it holds true for every label at every node simultaneously.

Let Λ_i^k denote the set of labels attached to node i at iteration k. Thus we have the iteration: begin with a label set Λ_i^0 at each node i and discard those labels that are not consistent according to the constraints $\Lambda_{i,j}(\lambda, \lambda')$. After each iteration we have

$$\Lambda_i^{k+1} = \{\lambda \in \Lambda_i^k \quad | \quad \forall j \in \eta(i) \quad \exists \lambda' \in \Lambda_j^k \quad \text{s.t.} \quad \Lambda_{i,j}(\lambda, \lambda')\}. \tag{26.2}$$

Convergence occurs when $\Lambda_i^{k+1} = \Lambda_i^k \quad \forall i$.

A different notation for the above operations can be introduced with characteristic functions. Let $p_i(\lambda)$ be one when λ is associated with node i, and zero otherwise; i.e.,

$$\hat{p}_i(\lambda) = \begin{cases} 1 & \lambda \to i \\ 0 & \text{otherwise} \end{cases} \tag{26.3}$$

Similarly, for the constraints let

$$\hat{R}_{i,j}(\lambda, \lambda') = \begin{cases} 1 & \lambda \to i \quad \text{and} \quad \lambda' \to j \\ 0 & \text{otherwise} \end{cases} \tag{26.4}$$

Now, we can write two formulas with these characteristic functions. First,

$$\hat{p}_i^{k+1} = \prod_{j \in neigh(i)} \max_{\lambda' \in \Lambda_j} \hat{R}_{ij}(\lambda, \lambda') \hat{p}_j^k(\lambda') \tag{26.5}$$

Notice how the max operation selects a single label (from perhaps many) at neighbor j, and how the product guarantees that, for $\hat{p}_i^{k+1} = 1$ there is consistency; i.e., $\hat{R}_{i,j}(\lambda, \lambda') = 1$.

Second, if we fix a label at each node in the graph but i, we have a measure of how much the labeling supports λ at i:

$$S_i(\lambda) = \sum_j \sum_{\lambda'} \hat{R}_{ij}(\lambda, \lambda') \hat{p}_j^k(\lambda') \tag{26.6}$$

This concept of support will be important because, for each λ

$$S_i(\lambda) \geq S_i(\lambda') \quad \forall \lambda' \in \Lambda_i \quad \forall i. \tag{26.7}$$

Inequalities related to this one will be at the foundation of consistency.

Penrose, L. S. and Penrose, R. "Impossible Objects: A Special Type of Visual Illusion." Brit. J. Psychology 49, 31-33, 1958.

26.5 Continuous Constraint Satisfaction

The next step was to generalize the characteristic functions from the set $\{0, 1\}$ to the interval $[0, 1]$. The result was interpreted as the probability $p_i(\lambda)$ that label λ should be associated with node i, with

$$\sum_{\lambda} p_i(\lambda) = 1 \quad \forall i.$$

The constraints were generalized as well. Compatibility functions $r_{i,j}(\lambda, \lambda')$ were postulated between label λ at position i and label λ' at position j such that increasingly positive values represent stronger compatibility. Intuitively, the idea was that, with $-1 \leq r_{i,j}(\lambda, \lambda') \leq 1$, $r_{i,j}(\lambda, \lambda') \approx 1$ implied that have label λ' at j was strongly consistent with λ at i, while $r_{i,j}(\lambda, \lambda') \approx -1$ implied that λ' at j was strongly **in**consistent with λ at i. Interpreting these compatibilities has been controversial, as we shall discuss later.

In any case, the challenge was to define a form of context for summarizing neighboring labels' influences on a given label at a given node. Clearly the compatibilities determine whether that influence should be positive or negative, strong or weak; and naturally, that influence should be weighted by the probability of the neighboring label. The following form was chosen:

$$q_i^k(\lambda) = \sum_j \sum_{\lambda'} r_{ij}(\lambda, \lambda') p_j^k(\lambda') \tag{26.8}$$

The $q_i^k(\lambda)$ provide a continuous summary of local context; it is positive if those labels with high probability are consistent, and negative if those labels are inconsistent. Thus positive contextual support for a label raises its probability, and negative contextual support lowers it. Iteration is designed to select the most consistent labeling.

After discounting linear updating rules for $p(\lambda)$, Rosenfeld, Hummel, and Zucker argued for the following nonlinear updating formula[1]:

$$p_i^{k+1}(\lambda) = \frac{p_i^k(\lambda)(1 + q_i^k(\lambda))}{\sum_\lambda p_i^k(\lambda)(1 + q_i^k(\lambda))}. \tag{26.9}$$

[1]In the original formulas there were weighting terms over positions, but since these can be absorbed into the $r_{ij}(\lambda, \lambda')$ we do not include them.

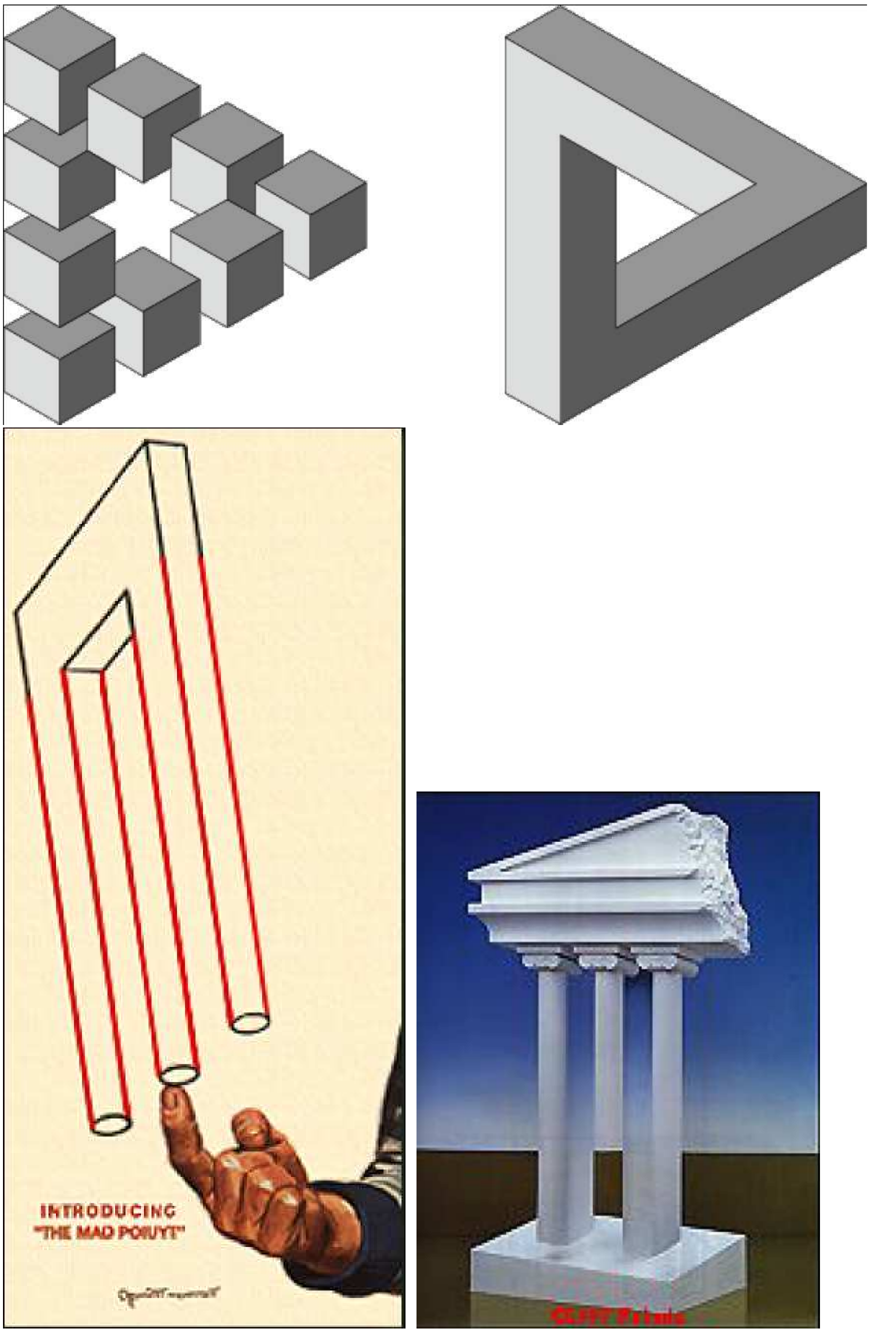

Figure 26.5: Impossible figures illustrating that consistency on the order of the figure is not required. (top) Penrose impossible triangle. The possible global results are given by a search over labelings; the structure of these labelings depends on the order of the constraints. Weisstein, Eric W. "Penrose Triangle." From MathWorld–A Wolfram Web Resource. http://mathworld.wolfram.com/PenroseTriangle.html (bottom) The impossible trident http://psylux.psych.tu-dresden.de/i1/kaw/diverses%20Material/www.illusionworks.com/html/impossible_trident.h

26.6 Relaxation Labeling Theory

(from Ohad)

Following early studies on contextual constraints in the interpretation of line drawings [?, ?], it has become widely acknowledged that interpretation of sensory data is highly unreliable unless carried out in some non-local context. Relaxation labeling [?, ?, ?, ?] is a formal computational framework for doing exactly that. Closely related to popular models of neural networks [?, ?] and polymatrix games [?] , it involves representing the interpretation problem as the assignment of labels to nodes in a graph whose edges represent the contextual structure.

Let $I = \{i \mid i = 1..n\}$ be a set of nodes, each of which may take any label λ from the set Λ. Let $p_i(\lambda)$ denote the probability, or *confidence* in the assignment of label λ to node i. In general, this measure needs to satisfy two constraints:

$$p_i(\lambda) \geq 0 \quad \forall i, \lambda \qquad \text{and} \qquad \sum_{\lambda \in \Lambda} p_i(\lambda) = 1 \quad \forall i.$$

Of all possible assignments (henceforth denoted as the space $\mathbb{K}$), those which satisfy $p_i(\lambda) \in \{0, 1\} \quad \forall i \in I, \forall \lambda \in \Lambda$ are called *unambiguous assignments* and assign a unique label to each node. The role of relaxation labeling is to start from a given, typically ambiguous labeling assignment, and iteratively change it toward a "better" one, where better refers to the degrees of ambiguity and consistency of the assignment of labels at different nodes.

The fundamental mechanism by which contextual information is propagated in a relaxation labeling network is a *compatibility function* $r_{ij}(\lambda, \lambda')$ which quantifies the contextual information conveyed by label λ' at node j about label λ at node i. In conjunction with the confidence measure p_i, the *contextual support* that a label λ at node i received from it's neighborhood is (almost exclusively) defined to be:

$$s_i(\lambda) = s(i, \lambda; \bar{r}, \bar{p}) = \sum_{j=1}^{n} \sum_{\lambda'=1}^{m} r_{ij}(\lambda, \lambda') p_j(\lambda'), \tag{26.10}$$

which can be viewed as a sum of all compatibilities weighted by the confidences. In the same spirit, we can average the supports themselves to yield a scalar measure of consistency across the entire network, or what is traditionally called the *average local consistency*

$$A(\bar{p}) = \sum_i \sum_\lambda p_i(\lambda) s_i(\lambda) = \sum_i \sum_\lambda \sum_j \sum_{\lambda'} p_i(\lambda) r_{ij}(\lambda, \lambda') p_j(\lambda'). \tag{26.11}$$

Finally, it remains to clarify when an assignment $\bar{p}$ is *consistent*. Intuitively, this happens when the labels assigned at each node maximally agree with their context. Had we dealt with unambiguous assignments only, this could have been expressed as a maximization of support at all nodes simultaneously, i.e.,

$$s_i(\lambda) \geq s_i(\lambda') \quad \forall i \in I.$$

This constraint is extendible to general (i.e., ambiguous) assignments via the following set of inequalities:

$$\sum_{\lambda} p_i(\lambda) s_i(\lambda; \bar{p}) \geq \sum_{\lambda} \tilde{p}_i(\lambda) s_i(\lambda; \bar{p}) \quad \forall \tilde{p}_i \in \mathbb{K}, \quad \forall i \in I.$$

Using all this machinery, relaxation labeling maps inconsistent labeling to consistent ones via an iterative process. Here we follow the algorithm by Hummel and Zucker [**?**] which is described by the following confidence update rule:

$$p_i^{t+1}(\lambda) \leftarrow \Pi_{\mathbb{K}} \left[p_i^t(\lambda) + \delta s_i^t(\lambda) \right], \tag{26.12}$$

where $\Pi_{\mathbb{K}}$ is a projection operator that projects its argument on $\mathbb{K}$, and δ is a constant step size. A fundamental result of the theory of relaxation labeling states that for symmetric compatibilities $r_{ij}(\lambda, \lambda') = r_{ji}(\lambda', \lambda)$, this update rule constitutes a gradient ascent on $A(\bar{p})$ which converges to a consistent labeling while maximizing $A(\bar{p})$ locally. Furthermore, if $\bar{r}$ is not symmetric, this algorithm still converges to a consistent labeling.

26.6.1 Relaxation Across Columns

Putting the relaxation algorithm together with the columnar architecture is now natural. The easiest way is shown in Fig. 26.6. Let all of the distinct labels (vocabulary entries) be listed along a column, and the compatibilities implemented by axons running between columns.

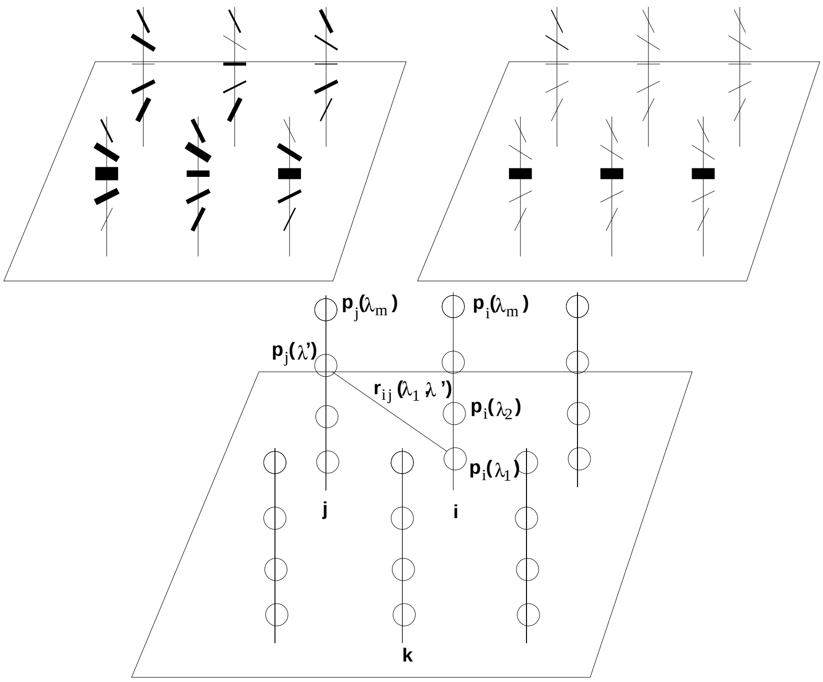

Figure 26.6: (top) Putative setup for running a relaxation labeling network within a columnar architecture. (top) probabilities for "edge" labels are indicated by thickness of lines; we seek to go from a noisy local estimate of edge orientation at a position to an unambiguous one (right). the weak results are elimated and only the strong ones survive. (bottom) Layout of the symbols in relaxation labeling theory to columnar model.