A DESIGN METHOD FOR RELAXATION LABELING APPLICATIONS

BY

Robert A. Hummel

- - - - - - - - - - - -

Technical Report No. 68
Robotics Report No. 11

March, 1983

Work on this paper has been supported in part by Office of Naval Research Grant N00014-82-K-0381, and by grants form the Digital Equipment Corporation, the Sloan Foundation, the System Development Foundation, and th IBM Corporation.

1. Introduction

In an earlier paper, a theory of consistency in ambiguous labelings of objects has been developed [1]. This theory allows the relaxation labeling algorithm to be interpreted as a method for finding consistent labelings and allows specific applications to be tailored in accordance with intended design goals. We begin by discussing the general class of problems to which this theory is applicable. We then outline the theory of consistency and the principal results implied by the relaxation labeling algorithm in conjunction with this theory. Finally, we discuss a design methodology for implementing applications.

2. Labeling Problems

The discrete labeling problem occurs when each of a finite number of objects are to be assigned a label chosen from a finite label set. Let $a_1, a_2, \ldots, a_n$ denote the n distinct objects, and $\{\lambda_1, \ldots, \lambda_m\}$ the set of possible labels. An unambiguous labeling is then an assignment mapping objects to labels. For example, if object a_i is labeled by λ_{ℓ_i}, then the integer map $i \rightarrow \ell_i$ can be used to define the labeling.

In practical systems, measurements are made to describe object a_i, and a most probable label is assigned independent of the labels assigned to neighboring objects. However, these decisions may be in error because of the myopia of a purely local viewpoint. Further, since the decision as to which label to assign to a particular object is fraught with uncertainty, information is discarded when we eliminate possible but improbable labels in favor of a single label at each object.

Consequently, it is useful to extend the notion of labelings to weighted labeling assignments. In this model, a nonnegative weight $p_i(\ell)$ is assigned for each label in the label set, at every object. The weight $p_i(\ell)$ denotes the confidence level that object a_i should be labeled by label λ_ℓ. These weights are normalized by the condition

$$\sum_{\lambda=1}^{m} p_i(\ell) = 1 .$$

The assignment $p_i(\ell) = 1$ signifies that object a_i is unambiguously labeled by label λ_ℓ, (see [2]). (The distribution of values $(p_i(1),\ldots,p_i(m))$ at object i makes the weights look very much like probabilities, but this viewpoint is unhelpful).

By concatenating assignment weights, we can form an assignment vector $\bar{p} = (\bar{p}_1,\ldots,\bar{p}_n)$, $\bar{p}_i = (p_i(1), p_i(2),\ldots,p_i(m))$. (Due to the normalization condition, there can be only one component with a value one at each object). We call the space of all possible assignment vectors $\bar{p}$ the assignment space IK. The set of unambiguous labelings can be viewed as a subset IK* IK, and is described by labelings in which all assignment components $p_i(\ell)$ are either 0 or 1. It is not hard to show that IK is the convex hull of IK* .

3. Support Functions

To apply the relaxation labeling method we must be given a set of objects and a label set, and also support functions $s_i(\ell;\bar{p})$, which are functions of the labeling assignment $\bar{p}$. A separate support function is given for each label at each object. Whereas $p_i(\ell)$ denotes the current confidence level that object a_i is label λ_ℓ, $s_i(\ell;\bar{p})$ denotes the support which the current mix of assignments of labels to objects in the vicinity of object a_i lends to the proposition that object a_i is label λ_ℓ. If the assignment $\bar{p}$ gives high confidence levels to labels which are compatible with label number ℓ at object number i, then $s_i(\ell;\bar{p})$ will be large and positive. If the local assignment of labels near object a_i leads to labels incompatible with λ_ℓ at a_i, then $s_i(\ell;\bar{p})$ will be negative.

In nearly all applications to date, the support functions have been linear functions of the labeling assignment $\bar{p}$. Thus there exist constants $r_{ij}(\ell,\ell')$ for every pair of object labels so that

$$s_i(\ell;\bar{p}) = \sum_{j=1}^{n} \sum_{\ell'=1}^{m} r_{ij}(\ell,\ell')p_j(\ell') .$$

In this model, $r_{ij}(\ell,\ell')$ represents the degree of compatibility between the decision to label a_i by λ_ℓ given that label $\lambda_{\ell'}$ has been assigned to a_j. This compatibility coefficient can be either positive or negative, depending on whether label $\lambda_{\ell'}$ at a_j lends positive or negative support to label λ_ℓ at a_i .

It is of course essential to know where the support functions come from. In first stating the theory, it is enough to suppose that the $s_i(\ell,\bar{p})$'s are "God-given." However, the principal task confronting the designer of a relaxation labeling application is the definition of formulae for computing the support functions. In Section 7, we show how the distinction between consistent and inconsistant labelings can be used to constrain the choice of support functions.

4. Consistency

A consistent assignment of labels to objects has historically been defined to mean an assignment which satisfies a system of logical constraints [3,4]. Use of support functions allows us to define consistency relative to a quantitative system of preferences, i.e. numerical weighted constraints. This extension applies to weighted labeling assignments as well as unambiguous labelings. In reviewing these notions, it is convenient to begin by discussing the situation for unambiguous assignments. Consistency is defined relative to the set of support functions. Suppose $\bar{p} \in IK^*$ represents an unambiguous label assignment which labels objects a_i , $i = 1,\ldots,n$ to labels $\lambda_{\ell_1},\ldots, \lambda_{\ell_m}$ respectively. We will say that the labeling $\bar{p}$ is consistent if

$$s_i(\ell_i;\bar{p}) \geq s_i(\ell;\bar{p}) \quad \text{for all } \ell, \quad \text{for } i = 1,\ldots,n \ .$$

That is, the unambiguous labeling is consistent if at each object the maximum support value is attained for the label actually assigned to that object. In other words, the distribution of support values is in agreement with the label assigned to every object. Note that support values are defined for every label at any given object, including those labels which are not assigned (have zero assignment weights), and that the calculation of the support values depends upon the assignment of weights at neighboring objects.

At first glance, it may seem trivial to find a consistent labeling. Many people suggest that assignments be changed so that the label with maximum support becomes the assigned label at each object. However, such a procedure fails because the support values will change when labeling assignments are changed. Instead, the consistency condition should be viewed as a coupled system of inequalities, expressing a "locking-in" property between the support values and the assignment weights.

Of course, many of the m^n possible unambiguous labelings will not be consistent. We cannot even guarantee, a priori, the existence of a consistent unambiguous labeling.

For weighted labeling assignments, we will say that $\bar{p} \in IK$ is consistent if

$$\sum_{\ell=1}^{m} p_i(\ell) s_i(\ell;\bar{p}) \geq \sum_{\ell=1}^{m} v_i(\ell) s_i(\ell;\bar{p}) \quad \text{for all } \bar{v} \in IK, \quad \text{for } i = 1,\ldots,n .$$

This definition is equivalent to the previous definition if $\bar{p}$ happens to be unambiguous. Simply stated, consistency occurs when the set of convex combinations of support values at a given object is maximized by the combination which chooses as its weights the assignment values, if this property holds true simultaneously at all objects.

5. The Relaxation Labeling Algorithm

A relaxation labeling process for problems formulated on the assignment space IK can use the following algorithm.

Step 1: Examine each object a_i, $i = 1,\ldots,n$, and using measurements about each object, obtain initial weighted assignments $p_i^0(\ell)$, $\ell = 1,\ldots,m$ signifying the confidence level for labeling a_i by λ_ℓ. Note: $\bar{p}^0 \in$ IK . Set $k = 0$.

Step 2: LOOP until convergence:

i. Compute $\bar{q}^k$, where $q_i^k(\ell) = s_i(\ell;\bar{p}^k)$

ii. Find the projection of $\bar{q}^k$:

$\bar{u}^k$:= Projection-Operator ($\bar{q}^k$, $\bar{p}^k$) .

iii. If $\bar{u}^k = 0$, then QUIT the loop.

iv. Otherwise, update $\bar{p}^k$ by

$\bar{p}^{k+1} := \bar{p}^k + \alpha_k \bar{u}^k$,

where α_k is chosen smaller than some predetermined step size (which may decrease as k increases) and so that $\bar{p}^{k+1} \in$ IK.

Increment k, $k := k+1$, and repeat loop.

Step 3: (Output of results) Assign label at each object by selecting the label with maximum current assignment weight. Output these labels.

The projection operator used in Step 2ii is defined by the following algorithm [5]:

1. Accept $\bar{q} = (\bar{q}_1,\ldots,\bar{q}_n) \in (\mathrm{IR}^m)^n$ and $\bar{p} = (\bar{p}_1,\ldots,\bar{p}_n) \in$ IK as input.
2. For $i = 1,\ldots,n$ do:

   a. Set $D := \{k \mid p_i(k) = 0\}$, $S := \emptyset$

b. LOOP until a quit is executed

i. Set ns := #S

ii. Set $t := (\sum_{k \notin S} q_i(k))/(m-ns)$

iii. $S := \{k \varepsilon D | q_i(k) < t\}$

iv. If #S = ns, then Quit, otherwise repeat loop.

c. Set $u_i(k) = q_i(k) - t$ for $k \notin S$,

$u_i(k) = 0$ for $k \varepsilon S$.

3. Normalize $\bar{u}$ by

$\bar{u} := \bar{u}/\|\bar{u}\|$ if $\bar{u} \neq 0$,

$\bar{u} := 0$ otherwise .

4. Return the vector $\bar{u}$.

The third step may be expensive. In many applications, the individual subvectors $\bar{u}_i$ are normalized independently, which is computationally much easier, and probably does not change the qualitative behavior of the relaxation labeling algorithm by much.

6. Theorems

The following results relate the notion of consistency and the relaxation labeling algorithm formulated above. Proofs are given in [1].

Result 1: If the relaxation labeling algorithm stops at $\bar{p}$, then $\bar{p}$ is a consistent labeling.

Result 2: If the support functions $s_i(\ell,\bar{p})$ are continuous in $\bar{p}$, then there always exists at least one consistent labeling.

Result 3: Suppose $\bar{p} \in$ IK* is an unambiguous consistent labeling. Suppose further that $\bar{p}$ is strictly consistent, in the sense that $s_i(\ell_i,\bar{p}) > s_i(\ell,\bar{p})$ for $\ell \neq \ell_i$, $i = 1,\ldots,n$, where $\bar{p}$ labels object number i by label number ℓ_i. Then there exists a neighborhood of $\bar{p}$ in IK such that if the iterates of the relaxation labeling process enter this neighborhood,or begin in it, then the process will converge to $\bar{p}$ (by a path of finite length) .

Result 4: Suppose that the support functions are linear in $\bar{p}$, and the corresponding coefficients $r_{ij}(\ell,\ell')$ are symmetric; i.e., $r_{ij}(\ell,\ell') = r_{ji}(\ell',\ell)$. Then the relaxation labeling process is equivalent to the method of steepest ascent for maximizing

$$\sum_i \sum_j \sum_\ell \sum_{\ell'} r_{ij}(\ell,\ell')p_i(\ell)p_j(\ell')$$

on IK .

7. Defining Support Functions

The proceding sections have summarized the results of [1]; in this section we present ideas not included in that paper. These relate to the fundamental pragmatic fact that the principal motivation for developing a theory of consistency in conjunction with the relaxation labeling algorithm is to be able to design support functions well adapted to particular applications.

Suppose that objects and label sets have been identified, and that formulae for support functions are required. The following method is suggested.

Certain patterns of unambiguous labelings can be identified as consistent labelings. Suppose, for example, that an unambiguous labeling $i \to \ell_i$, denoted succinctly by $\bar{\ell} = (\ell_1,\ldots,\ell_n)$, is to be viewed as consistent. Let $\bar{p}$ be the corresponding unambiguous assignment vector. Then we want the inequalities $s_i(\ell_i,\bar{p}) > s_i(\ell;\bar{p})$, to be satisfied. If the support functions are to be designed as linear functions of $\bar{p}$, these conditions can be written as

$$\sum_j r_{ij}(\ell_i,\ell_j) > \sum_j r_{ij}(\ell,\ell_j), \quad \text{all } \ell, \text{ all } i \ .$$

Suppose that $\bar{\ell}^1$, $\bar{\ell}^2,\ldots,\bar{\ell}^N$ are N distinct patterns of labelings which are all deemed to be consistent. Then we want

$$\sum_j r_{ij}(\ell_i^k, \ell_j^k) > \sum_j r_{ij}(\ell,\ell_j^k)$$

to be satisfied for all ℓ, all i, k = 1,...,N .

These conditions may constitute a large number of linear inequalities in the set of variables $r_{ij}(\ell,\ell')$. The system of inequalities may have no nontrivial solution,* in which case it is impossible to design linear support functions with $\bar{\ell}^1,\ldots,\bar{\ell}^N$ as consistent labeling patterns. However, if the system has a nonempty solution set, then any assignment of values to the $r_{ij}(\ell,\ell')$'s satisfying the inequalities is called a feasible solution. In this case, linear programming methods (such as the simplex method) can be used to find feasible solutions.

The set of feasible solutions forms a cone. That is, any set of values for the $r_{ij}(\ell,\ell')$'s can be multiplied by a uniform positive scalar, and the resulting set will still provide a feasible solution. This is clear from the form of the inequalities. Of course, such multiplication will not affect the relaxation labeling algorithm.

If the interior of the set of feasible solutions is nonempty, then there exist strictly feasible solutions. A strictly feasible solution is an assignment of values to the $r_{ij}(\ell,\ell')$'s such that the system of inequalities hold strictly. That is,

$$\sum_j r_{ij}(\ell_i^k, \ell_j^k) > \sum_j r_{ij}(\ell,\ell_j^k), \text{ all } \ell \neq \ell_i^k, \text{ all } i,\ k = 1,\ldots,N.$$

If a strictly feasible solution is chosen, then the initially given labeling patterns $\bar{\ell}^1,\ldots,\bar{\ell}^N$ will correspond to strictly consistent

*The trivial solution $r_{ij}(\ell,\ell') = 0$ is uninteresting because it classifies all labelings as consistent.

labelings. According to Result 3, these unambiguous labeling patterns will then be local attractors of the relaxation labeling process.

It may well happen that a strictly feasible solution will yield other consistent unambiguous labelings not represented in the design pattern set. This may be undesirable, and will require a search for a feasible solution which minimizes the problem of spurious consistent labelings. An example in the next section illustrates a method for accomplishing this search.

8. Examples

It is instructive to consider a few simple examples.

Suppose that the graph of objects is given by a hexagonal grid, so that each object is equidistant from six neighbors. Consider the simplest case of two labels. We suppose that the following local patterns are consistent:

1 1	2 2	1 2	2 1
1 1 1	2 2 2	1 1 2	2 2 1
1 1	2 2	1 1	2 2

Further, we assume that the relationship of consistency is "isotropic", that is, a rotation of a consistent labeling is consistent.

Note that labels 1 and 2 are treated symmetrically in this example, so in searching for coefficients r_{ij} we can assume that $r_{ij}(1,1) = r_{ij}(2,2)$ and $r_{ij}(1,2) = r_{ij}(2,1)$. Further, since each neighbor is equidistant and direction is unimportant, the r_{ij}'s are independent of i and j as long as i and j are distinct neighboring objects. (We assume that the $r_{ii}(\ell,\ell')$'s are zero.)

Thus only two parameters will enter into the coefficients r_{ij} that we seek: for i and j distinct neighbors, we set $a = r_{ij}(1,1)$ and $b = r_{ij}(1,2)$. Applying the conditions for strict consistency to each of the patterns listed above leads to the single condition $a > b$.

What other unambiguous local patterns are consistent in this scheme? Let n_1 be the number of 1-labels among the six neighbors of a central object, and n_2 be the number of 2-labels. Then the support for label 1 is $s(1) = an_1 + bn_2$ and the support for label 2 is $s(2) = an_2 + bn_1$. A local pattern with a central object labeled with a "1" is strictly consistent if $s(1) > s(2)$, i.e., $n_1 > n_2$. (Note that $a-b > 0$). Similarly, "2" is consistent for the central object of a local pattern if $n_2 > n_1$.

Remarkably, the same set of patterns are consistent whatever the values of a and b, as long as $a > b$. This confirms empirical evidence of the robustness of simple relaxation labeling applications to modifications of the compatibilities.

A global unambiguous labeling will be strictly consistent in this model if every object is labeled with the majority label as voted by the six neighbors. Since the condition holds at every object, it is not hard to see that strictly consistent labelings consist of strips of 1's and 2's with straight parallel interfaces between the regions. These straight line interfaces must occur in one of the six principal directions of the hexagonal grid.

Clearly, this example is too simple to be of assistance in any practical application. We will consider a slightly more complicated situation to illustrate the method. However, practical situations will generally be much more complex than either of our examples.

As in the previous example, consider a hexagonal array of objects, this time with three labels. The labels are "1" and "2" which we can regard as "region types", and label "3" to denote "edge between 1 and 2's". We regard a local pattern of either constant 1's or constant 2's as a consistent labeling:

```
 1 1        2 2
1 1 1      2 2 2
 1 1        2 2
```

A region of 1's separated from 2's by a line of 3's is consistent:

```
 3 2
1 3 2
 1 3   .
```

For the same reason

```
 1 3               2 3
1 1 3     and     2 2 3
 1 1               2 2  .
```

are regarded as consistent. As before, we treat labels 1 and 2 symmetrically, and assume isotropy.

Set

$$a = r(1,1) = r(2,2)$$

$$b = r(1,2) = r(2,1)$$

$$c = r(3,3)$$

$$d = r(1,3) = r(2,3)$$

$$e = r(3,1) = r(3,2) .$$

Each pattern yields two inequalities to constrain the five parameters. From the constant patterns, we deduce

$$a > b$$

$$a > e .$$

From the pattern with two 1's and two 2's, $4e + 2c > 2a + 2b + 2d$, i.e.,

$$2e + c > a + b + d,$$

and from the pattern with only two 3's, $4a + 2d > 2c + 4e$, i.e.,

$$2a + d > 2e + c .$$

Combining, we have

$$a > b,\ a > e,\ \text{and}$$

$$(a-e) + (b-e) < c-d < 2(a - e).$$

It is easy to see that feasible solutions of these equations exist and can be readily constructed. In particular, choose any positive value for a', then choose $b' < a'$, and e arbitrary. Set $a = a'+e$, $b = b'+e$, and finally choose a value for $c-d$ between $a'+ b'$ and $2a'$. The values for c and d can be selected so that the difference is the specified value.

The designated patterns, and all their rotations, will be strictly consistent under the compatibilities of any feasible solution. We will now try to select a feasible solution which gives rise to as few other consistent labeling patterns as possible.

Let n_1, n_2 and n_3 denote the number of neighbors of a central object of a local hexagonal cell having labels 1, 2, and 3 respectively. The label "1" at the central object is part of a consistent pattern if

$$an_1 + bn_2 + dn_3 > dn_1 + an_2 + dn_3$$

$$\text{and } an_1 + bn_2 + dn_3 > en_1 + en_2 + cn_3 \text{ .}$$

From the first inequality, and the fact that $a > b$, we obtain

$$n_1 > n_2 \text{ .}$$

Using the fact that $n_1 + n_2 + n_3 = 6$, the second inequality becomes

$$[(c-d) + (b-e)]n_3 < (a-b)n_1 + 6(b-e) \text{ .}$$

In a similar way, it can be shown that "3" is a consistent label for the central object of a local pattern if

$$[(c-d) + (b-e)]n_3 > (a-b)\cdot \max(n_1,n_2) + 6(b-e) .$$

Let us arbitrarily choose a = 1, and b = -1. Then e = r(3,1) = 0 makes sense since "3" and "1" can co-occur. Having chosen a,b, and e, then 0 < c-d < 2. Suppose we choose c-d = 1. Then applying the conditions above, a central "1" is consistent if

$$n_1 > n_2 \quad \text{and} \quad n_1 > 3 .$$

For a "3" to be consistent, it suffices to have

$$\max(n_1,n_2) < 3 .$$

Thus a pattern with a "1" or "2" label in the center is consistent if there are four or more of the same label type in the neighborhood. A central "3" is consistent as long as there are two or fewer "1"'s and two or fewer "2"'s in the neighborhood.

This seems to be reasonable behavior. But note that the pattern

1 2

1 x 2

1 1

is consistent with x = 1 under the above choice of values. We would prefer the support for x = 3 to be higher than the support for x = 1 to justify the interpretation of label "3" as "edge." Thus we would like $6e > 4a + 2b$.

Having chosen $a = 1$, $b = -1$, we now see that $e > 1/3$ is desirable. Let $e = 2/3$, whence $-2/3 < c-d < 4/3$. By varying the value of one parameter c-d, different behavior of the patterns of consistency can be selected. For example, suppose we choose c-d to be 1/2. Then by applying the previous inequalities and a simple case analysis, the following statements can be made for this assignment of weights:

A "1" label is consistent only if there are four or more "1" labels among the six neighbors, and no "2" labels.

A "3" label is consistent if there are three or more "3" labels among the neighbors, or if there are four of label type "1" or "2", and at least one of the other type.

This example illustrates how an initial assignment of values to the $r_{ij}(\ell,\ell')$'s obtained as a feasible solution can be refined to give more desirable behavior by the addition of a constraint. In this case, the additional inequality arises when we decide to reject a spurious consistent pattern, and the statement that a particular label should have greater support than the otherwise consistent label.

9. Acknowledgement

Relaxation labeling, the foundational theory and notion of consistency, and the idea of applying the theory to obtain design criteria, were developed in collaboration with Professor Steven Zucker.

References

1. Hummel, R.A. and S.W. Zucker, "On the Foundations of Relaxation Labeling Processes," IEEE-PAMI, to appear 1983.

2. Rosenfeld, A., R. Hummel, and S. Zucker, "Scence Labeling by Relaxation Operations," IEEE-Trans. on Systems, Man, Cybernetics 6, 1976, p. 420.

3. Mackworth, A.K., "Consistency in Networks of Relations," Artificial Intelligence 8, 1977, p. 99.

4. Haralick, R.M. and L. Shapiro, "The Consistent Labeling Problem: Part I," IEEE-PAMI 1, 1979, p. 173.

5. Mohammed, J.L, R.A. Hummel, and S.W. Zucker, "A Gradient Projection Algorithm for Relaxation Methods," IEEE-PAMI, to appear 1983.