A SIMPLEX LIKE ALGORITHM FOR RELAXATION LABELING PROCESS

Xinhua Zhuang, Robert M. Haralick, and Hyonam Joo

Machine Vision International
Ann Arbor, Michigan 48104 USA

Abstract

In this paper a simplex like algorithm is developed for the relaxation labeling process. The algorithm is simple and has a fast convergence property which can be summarized by a "one more step theorem." The algorithm is based on fully exploiting the linearity of the variational inequality and the linear convexity of the consistent labeling search space.

I. Introduction

R.A. Hummel and S.W. Zucker (1983) developed a theory to explain what relaxation labeling accomplishes. The theory is based on an explicit new definition of consistency in terms of a variational inequality and leads to a relaxation algorithm with an updating formula which uses a projection operator.

By fully exploiting the linearity of the variational inequality and the linear convexity of consistent labeling search space, we obtain an essential characterization of a consistent labeling (see Section II). An efficient simplex like algorithm is developed in Section III. The convergence of the algorithm is explored in Section IV (see Theorem 1-2). Theorem 1 carries the name 'one more step theorem' which indicates the algorithm takes the shortest path. A comparison with Rosenfeld et al. consistent labeling definition is made in Section V. The results which are given in the final section, section VI, verify the theory and algorithm developed in the paper.

II. Consistent Labeling: Definition and Characterization

A consistent labeling problem has units each of which has an unknown true label. There are n units, denoted by $U_1, ..., U_n$, and m possible labels, denoted by $L_1, ..., L_m$. Associated with each U_i $(i = 1, ..., n)$ will be a set of m numbers $p_1(i), ..., p_m(i)$ constituting a labeling distribution:

$$p_1(i) \geq 0, ..., p_m(i) \geq 0, \tag{1}$$

$$\sum_{j=1}^{m} p_j(i) = 1. \tag{2}$$

For abbreviation, we let

$$p(i) \triangleq (p_1(i) ..., p_m(i)), \qquad i = 1, ..., n, \tag{3}$$

and simply call the $(1 \times m)$ row vector $p(i)$ a labeling distribution for unit U_i.

Between label assignments there are consistency constraints. Let a real number $r(i,j;h,k)$ represent how the label L_k at the unit U_h influences the label L_j at the unit U_i: If the unit U_h having the label L_k lends a high support to the unit U_i having the label L_j, then $r(i,j;h,k)$ should be large and positive. If constraints are such that the unit U_h having the label L_k means that the label L_j at the unit U_i is highly unlikely, then $r(i,j;h,k)$ should be small. No specific restrictions are placed on the magnitude of $r(i,j;h,k)$. However, we do require that

$$r(i,j;h,k) \leq r(i,j;i,j). \tag{4}$$

Formally, we define the support on the unit U_i having the label L_j, $\{U_i,\ L_j\}$, from the unit U_h having the label L_k with a labeling distribution component $p_k(h)$, i.e. $\{U_h,\ L_k,\ p_i(h)\}$, by $r(i,j;h,k)p_k(h)$; the support on $\{U_k,\ L_j\}$ from the unit U_h having a labeling distribution $p(h)$, i.e. $\{U_h,\ p(h)\}$, by $\sum_{k=1}^{m} r(i,j;h,k)p_k(h)$; the support on $\{U_i,\ L_j\}$ from n labeling distributions $[p(1), ..., p(n)]$, or $P \triangleq [p(1), ..., p(n)]$, by $q_j(i;P)$:

$$q_j(i;P) \triangleq \sum_{h=1}^{n} \sum_{k=1}^{m} r(i,j;h,k)p_k(h). \tag{5}$$

Furthermore, we define the support on unit U_i having the label L_j with another labeling distribution component $v_j(i)$, i.e. $\{U_i,\ L_j,\ v_j(i)\}$, from P by $q_j(i;P)v_j(i)$ the support on the unit U_i having another labeling distribution $v(i)$, i.e. $\{U_i,\ v(i)\}$ from P by the inner product $(q(i;P),\ v(i))$ in the m-dimensional Euclidean space E_m where

$$q(i;P) \triangleq (q_1(i;P), ..., q_m(i;P)). \tag{6}$$

Finally we define the support on another set of n labeling distributions $[v(1), ..., v(n)]$, or $V \triangleq [v(1), ..., v(n)]$, by the inner product $(q(P),\ V)$ in the nm-D Euclidean space. E_{nm} where

$$q(P) \triangleq [q(1;P), ..., q(n;P)], \tag{7}$$

$$(q(P),\ V) = \sum_{i=1}^{n}(q(i;P),\ v(i)). \qquad (8).$$

For convenience we simply call each of P and V a labeling. Thus, the support on the labeling V from the labeling P is represented by the inner product $(q(P),\ V)$.

A set of n labeling distributions $\{p(1), ..., p(n)\}$ is called unambiguous if each of n units is assigned a unique label, that is, for each i, $1 \leq i \leq n$, all $p_j(i)$'s $(j = 1, ..., m)$ are zero except one which is 1. Hummel and Zucker first define a consistency concept of an unambiguous labeling, then by analogy they define a consistency concept of an ambiguous labeling. According to their definition, n labeling distributions $p(1), ..., p(n)$ comprise a consistent labeling if for various n labeling distributions $v(1), ..., v(n)$ there hold the following variational inequalities:

$$(q(i;P),\ v(i) - p(i)) \leq 0, \qquad i = 1, ..., n, \qquad (9)$$

or equivalently

$$(q(i;P),\ v(i)) \leq (q(i;P),\ p(i)), \qquad i = 1, ..., n. \quad (10)$$

In other words, P is a consistent labeling iff for each i, $i = 1, ..., n$, $p(i)$ maximizes $(q(i;P),\ v(i))$ as $v(i)$ varies over the simplex K in E_m (see Eq.(13)). Thus a consistent labeling P gives the support in favor of itself or discriminates against any other labelings since

$$\begin{aligned}(q(P),\ V) &= \sum_{i=1}^{n}(q(i;P),\ v(i)) \\ &\leq \sum_{i=1}^{n}(q(i;P),\ p(i)) \\ &= (q(P),\ P).\end{aligned}$$

Conversely, if a labeling P gives the support in favor of itself, i.e., for any other labeling V, it holds that

$$(q(P),\ V) \leq (q(P),\ P), \qquad (11)$$

then the labeling P is consistent, i.e. for each i, $i = 1, ..., n$ Eq.(10) holds, since letting each $v(h)$ equal $p(h)$ except $v(i)$ which could be arbitrary, Eq.(11) will imply Eq.(10), as easily verified.

Therefore, a consistent labeling P could also be defined by the single variational inequality, Eq.(11). In other words, P is a consistent labeling iff P maximizes $(q(P),\ V)$ as V varies over K^n (see Eq.(15)).

Let $e_1, ..., e_m$ be m standard basis vectors in E_m. Let

$$K_0 \triangleq \{e_1, ..., e_m\}, \qquad (12)$$

$$K \triangleq \left\{\sum_{j=1}^{m} u_j e_j : u_j \geq 0,\ \sum_{j=1}^{m} u_j = 1\right\}, \qquad (13)$$

$$K_0^n \triangleq \prod_{i=1}^{n} K_0, \qquad (14)$$

$$K^n \triangleq \prod_{i=1}^{n} K \qquad (15)$$

Then $K(K^n)$ is a linear convex set in $E_m(E_{nm})$ and $K_0(K_0^n)$ the set of vertices of $K(K^n)$. The set K takes a specific name 'simplex' in Topology and Linear Programming. It is clear that $q(P)(q(i;P))$ defines a linear transformation:$K^n \rightarrow E_{nm}(E_m)$ and $q_j(i;P)$ a linear functional: $K^n \rightarrow E_1$. The inner product $(q(P),\ V)$ defines a bilinear functional: $K^n \times K^n \rightarrow E_1$ and the inner product $(q(i;P),\ v(i))$ a bilinear functional: $K^n \times K \rightarrow E_1$.

Hummel and Zucker call a labeling P strictly consistent if for each $v(i) \in K$, $v(i) \neq p(i)$, it holds that

$$(q(i;P),\ v(i)) < (q(i;P),\ p(i)), \quad i = 1, ..., n. \quad (16)$$

In other words, P is a strictly consistent labeling iff for each i, $i = 1, ..., n$, $p(i)$ is a unique maximal point of $(q(i;P),\ v(i))$ as $v(i)$ varies over K. Similarly, it could be proved that a labeling P is strictly consistent iff for each $V \in K^n$, $V \neq P$, it holds that

$$(q(P),\ V) < (q(P),\ P). \qquad (17)$$

In other words, P is strictly consistent iff P is a unique maximal point of $(q(P),\ V)$ as V varies over K^n.

The consistency condition suggests that to find a consistent labeling $P = [p(1), .., p(n)]$ with $p(i) = (p_1(i), ..., p_m(i))$ we need first to consider

$$\max_{v(i)\in K}(q(i;P),\ v(i)), \qquad i = 1, ..., n. \qquad (18)$$

Each maximum will be reached at vertices of the simplex K since each inner product $(q(i;P),\ v(i))$ is linear with respect to $v(i)$ and the search space K a linear convex set. Denote the maximal vertex set by $M_0(i;P)$. That is

$$M_0(i;P) \triangleq \{e_j :\ (q(i;P),\ e_j) = \max_{1\leq k\leq m}(q(i;P),\ e_k)\}. \qquad (19)$$

Let $M(i;P)$ be the linear convex set having $M_0(i;P)$ as its vertex set. Then it is clear that $M(i;P)$ comprises a face of K and represents the maximal point set. That is

$$\begin{aligned}M(i;P) = \{u(i) \in K :\ &(q(i;P),\ u(i)) \\ &= \max_{v(i)\in K}(q(i;P),\ v(i))\}.\end{aligned} \qquad (20)$$

From Eq.(19), it is easy to derive that

$$M_0(i;P) = \{e_j :\ q_j(i;P) = \max_{1\leq k\leq m} q_k(i;P)\}, \qquad (21)$$

and hence

$$M(i;P) = \left\{\sum_{j=1}^{m} u_j e_j \in K :\ u_j = 0 \text{ if } e_j \notin M_0(i;P)\right\}. \qquad (22)$$

Now it becomes apparent that P is a consistent labeling if and only if

$$p(i) \in M(i; P), \qquad i = 1, ..., n. \tag{23}$$

And P is a strictly consistent labeling if and only if

$$M(i; P) = M_0(i; P) = \{p(i)\}, \qquad i = 1, ..., n. \tag{24}$$

In the latter case each $p(i)$ must be a vertex of K and hence a strictly consistent labeling is unambiguous.

Let

$$M_0(P) = \prod_{i=1}^{n} M_0(i; P), \tag{25}$$

$$M(P) = \prod_{i=1}^{n} M(i; P). \tag{26}$$

Then we could characterize a consistent labeling P by:

$$P \in M(P), \tag{27}$$

and a strictly consistent labeling P, which must be a vertex of K^n, by:

$$M(P) = M_0(P) = \{P\}. \tag{28}$$

From a practical point of view, strictly consistent labelings are our favorite because they are unambiguous and isolated, the latter will be explained in the next section.

III. A Simplex Algorithm

Similar to the approach in Linear Programming, our reasoning first leads to the maximal vertex set. $M_0(P) \subset K_0^n$, and then the maximal point set, $M(P) \subset K^n$, formed by $M_0(P)$, where each $M(i; P)$ comprises a face of the simplex, K. If $P \in M(P)$, then P is a consistent labeling. If not, what is the next candidate consistent labeling to choose? Suppose $W(P) \triangleq [w(1; P), ..., w(n; P)]$ is the orthogonal projection of P onto $M(P)$. That is,

$$W(P) \in M(P), \|W(P) - P\| = \min_{v \in M(P)} \|V - P\|. \tag{29}$$

It is apparent that $W(P)$ is uniquely determined by P and each $w(i; P)$ is the orthogonal projection of $p(i)$ onto $M(i; P)$, i.e.,

$$w(i; P) \in M(i; P), \quad \|w(i; P) - p(i)\| = \min_{v(i) \in M(i; P)} \|v(i) - p(i)\| \tag{30}$$

A consistent labeling P could be characterized as:

$$P = W(P) \quad \text{or} \quad p(i) = w(i; P), \quad i = 1, ..., n. \tag{31}$$

When $P \notin Z$, it seems reasonable to choose $W(P)$ as the next candidate consistent labeling.

Let

$$w_j(i; P) = \begin{cases} p_j(i) = \sum_{k: e_k \notin M_0(i;P)} p_k(i)/\#M_0(i; P), & e_j \in M_0(i; P), \\ 0, & \text{otherwise.} \end{cases} \tag{32}$$

$$w(i; P) = (w_1(i; P), ..., w_m(i; P)).$$

$$W(P) = [w(1; P), ..., w(n; P)].$$

Then $w(i; P)(W(P))$ defined by Eq.(32) is the orthogonal projection of $p(i)(P)$ onto $M(i; P)(M(P))$. It is easy to see that

$$w_j(i; P) \geq 0, \ = 0 \text{ if } e_j \notin M_0(i; P).$$

$$\sum_{j=1}^{m} w_j(i; P) = \sum_{j: e_j \in M_0(i;P)} p_j(i) + \sum_{k: e_k \notin M_0(i;P)} p_k(i) = 1.$$

and for any $v(i) \in M(i; P)$

$$(v(i),\ w(i; P) - p(i)) = \sum_{j: e_j \in M_0(i;P)} v_j(i)[w_j(i; P) - p_j(i)]$$

$$= \left(\sum_{j: e_j \in M_0(i;P)} v_j(i) \right) \left(\sum_{k: e_k \notin M_0(i;P)} p_k(i) \right) / \#M_0(i; P)$$

$$= \sum_{k: e_k \notin M_0(i;P)} p_k(i)/\#M_0(i; P).$$

which is independent of $v(i)$. Therefore, $w(i; P)$ belongs to $M(i; P)$ and comprises the unique orthogonal projection of $p(i)$ onto $M(i; P)$.

Now we are able to summarize the algorithm:

Step 1. Set P^1.
Step 2. Set $k = 1$.
Step 3. Compute $M_0(P^k)$.
Step 4. Compute $P^{k+1} = W(P^k)$.
Step 5. If $(P^{k+1} = P^k)$ Stop.
Step 6. Set $k = k + 1$.
Step 7. Go To Step 3.

The next section is devoted to the convergence discussion.

IV. Convergence Discussion

As seen, the proposed algorithm is simple and easily implementable. It has also nice convergence properties since the linearity of variational inequalities and linear convexity of the consistent labeling search space are exploited. The following Theorem 1 is something similar to the local convergence theorem by Hummel and Zucker, but it is a little bit nicer. It confirms that the algorithm finds the shortest path: When it starts with a point close to a strictly consistent labeling, only one more iteration is needed to reach the goal. Theorem

2 relates that any sequence produced by the algorithm, if it converges, must converge to a consistent labeling.

Theorem 1. (One More Step Theorem) Assume P^0 is a strictly consistent labeling. Then, when P^k is close to P^0, only one more iteration is needed to reach the goal, P^0. That is,

$$P^{k+1} = P^0 \tag{33}$$

Proof: Since P^0 is a strictly consistent labeling, $M(P)$ will consist of a single point, P^0, whenever $\|P - P^0\|$ is small, as argued before, Thus, when P^k is close to P^0, it holds that

$$M(P^k) = \{P^0\}.$$

which implies that

$$P^{k+1} = W(P^k) = P^0.$$

since the orthogonal projection of P^k onto $\{P^0\}$ equals P^0.

V. Comparison with Rosenfeld et al. (1976) Consistent Labeling Definition

Using the same notation as in Section II, the Rosenfeld et al. relaxation labeling update scheme is as follows:

$$p_j(i) := \frac{p_j(i)[1 + q_j(i;P)]}{\sum_{k=1}^{m} p_k(i)[1 + q_k(i;P)]} \tag{34}$$

$$j = 1, ..., m; \qquad i = 1, ..., n.$$

When $|r(i,j;h,k)| \ll$, their assumption, $|q_j(i;P)| < 1$, will be satified. A labeling P is consistent in Rosenfeld et al.'s sense if P is a fixed point of Eq.(34). An essential condition of a consistent labeling in Rosenfeld et al.'s sense is that for each $p_j(i) > 0$, $q_j(i;P)$ keeps constant, independent of j. Using this characterization, we could prove that a Hummel and Zucker's consistent labeling is a Rosenfeld et al.'s consistent labeling: Suppose P is a

Case	Initial distributions				Algorithm 1 after 25 iterations				Algorithm 2 after 1 iteration			
A	.25	.25	.25	.25	.27	.27	.23	.23				
	.25	.25	.25	.25	.27	.27	.23	.23				
	.25	.25	.25	.25	.27	.27	.23	.23				
B	.5	0	.5	0	.99	0	.01	0	1	0	0	0
	.5	0	.5	0	.99	0	.01	0	1	0	0	0
	.5	0	.5	0	.99	0	.01	0	1	0	0	0
C	.5	0	.5	0	.99	0	.01	0	1	0	0	0
	.4	0	.6	0	.91	0	.09	0	1	0	0	0
	.5	0	.5	0	.99	0	.01	0	1	0	0	0
D	.5	0	.5	0	1	0	0	0	1	0	0	0
	.3	0	.7	0	.19	0	.81	0	1	0	0	0
	.5	0	.5	0	1	0	0	0	1	0	0	0
E	.3	0	.7	0	.9	0	.1	0	1	0	0	0
	.3	0	.7	0	.9	0	.1	0	1	0	0	0
	.5	0	.5	0	1	0	0	0	1	0	0	0
F	.2	0	.8	0	.07	0	.93	0	1	0	0	0
	.3	0	.7	0	1	0	0	0	1	0	0	0
	.5	0	.5	0	1	0	0	0	1	0	0	0
G	.3	.2	.3	.2	.98	0	.02	0	1	0	0	0
	.3	.2	.3	.2	.98	0	.02	0	1	0	0	0
	.3	.2	.3	.2	.98	0	.02	0	1	0	0	0
H	.3	.2	.3	.2	1	0	0	0	1	0	0	0
	.25	.25	.25	.25	1	0	0	0	1	0	0	0
	.2	2	.4	.2	.11	0	.89	0	0	0	1	0
I	.5	0	.5	0	1	0	0	0	1	0	0	0
	.02	0	.98	0	0	0	1	0	0	0	1	0
	.5	0	.5	0	1	0	0	0	1	0	0	0

Fig. 1. Experimental result of the line labeling

Hummel and Zucker's consistent labeling. Then for each $i, i = 1, ..., n$, $p(i)$ belongs to $M(i; P)$, which means that for each $p_j(i) > 0$, $q_j(i; P) = \max_{1 \le k \le m} q_k(i; P)$, a constant independent of j. This completes the proof.

Since the Hummel and Zucker's consistent labeling set is nonempty, the Rosenfeld et al.'s consistent labeling set is nonempty, too.

VI. Experimental Results and Summary

The simple example of scene labeling considered by Rosenfeld et al. (see [3]) is used to verify the new relaxation algorithm developed in this paper. The problem is to label the line of a triangle shown in Fig. 1 in [3].

The compatibility of label λ on unit a_i with label λ' on unit a_j, $r_{ij}(\lambda, \lambda')$ is related to the function $r(i, \lambda; j, \lambda')$ in this paper as:

$$r(i, \lambda; j, \lambda') = d_{ij} \cdot r_{ij}(\lambda, \lambda')$$

where d's are constant coefficients. Then, the function $q_i^{(k)}(\lambda)$ which is the change in $p_i^{(k)}(\lambda)$ in the k^{th} iteration, where $q_i^{(k)}(\lambda)$ are the notation used in [3], is the same as the support function $q_\lambda(i; P^k)$ in the new algorithm. We use the same values for $f_{ij}(\lambda, \lambda')$ and d_{ij} as Rosenfeld et al. used in their example.

The problem is to label three units $U_i (i = 1, 2, 3)$ three sides of a triangle, with four labels $L_i (i = 1, ..., 4)$ the set of four line labels $\{+, -, \rightarrow, \leftarrow\}$ used by Waltz (see [4]). The behavior of the label distributions for the algorithm proposed by Rosenfeld et al. (Algorithm 1) and the one proposed in this paper (Algorithm 2) is illustrated in Fig. 1 for various initial labeling distributions. The row vector of each matrix in the figure represents the labeling distribution for each unit.

The first iteration using Algorithm 2 in Case A gives

0.5	0.5	0	0
0.5	0.5	0	0
0.5	0.5	0	0

and the second iteration in the same case gives

0	0	0.5	0.5
0	0	0.5	0.5
0	0	0.5	0.5

Afterwards the results oscillate. However, Algorithm 1 after 25 iterations gives

.27	.27	.23	.23
.27	.27	.23	.23
.27	.27	.23	.23

It seems both algorithms do not give a meaningful interpretation in Case A. In cases B, C, E, and G, both algorithms give the most probable interpretation. In case H, both algorithms give a less probable interpretation. In case I, both algorthms give the desired interpretation. In cases D and F, two algorithms give different interpretations. However Algorithm 2 gives the most probable interpretation In all cases except case A, Algorithm 2 takes only one interation to reach the goal in comparison with more than 25 iterations required by Algorithm 1.

References

(1) R. Hummel and S. Zucker, "On the Foudations of Relaxation Labeling Processes", *IEEE Trans. Pattern Recognition and Machine Intelligence*, Vol. PAMI-5, 1983, p. 267.

(2) A. Rosenfeld, R. Hummel, and S. Zucker, "Scene Labeling by Relaxation Operations", *IEEE Systems, Man, and Cybernetics*, Vol. SMC-6, 1976, p. 420.

(3) D.L. Waltz, "Generating Semantic Descriptions from Drawings of Scenes with Shadows", MIT Technical Report AI 271, Nov. 1972.