面向大数据集的密切关系传播算法改进

杨栋，郭平*
北京师范大学图形图像与模式识别实验室，北京，中国，100875
d.yang@ieee.org, pguo@ieee.org

【摘要】密切关系传播(AP)算法是一种基于图消息传递的新型鲁棒聚类算法，能自动确定类别数目且不受维数约束。但在处理密集大数据集时，其时间复杂度较高。现有的改进方法大都处理输入数据矩阵，而不是改进算法本身。本文对消息传递机制进行研究，并阐明了新算法能够保持原始 AP 算法的精度，且较大地降低时间复杂度。实验证明了该算法能够大幅降低原始 AP 算法在具有重复点的大数据集上的处理时间，同时性能优于目前已有的针对具有重复点的大数据集的 AP 算法，在某些数据集上保持甚至超越了原始 AP 算法的精度。

【关键词】密切关系传播算法；大数据集；消息传递

Improved Affinity Propagation Algorithm for Large Dataset

Dong Yang, Ping Guo*
Image Processing and Pattern Recognition Laboratory, Beijing, China
d.yang@ieee.org, pguo@ieee.org

Abstract: Affinity propagation (AP) algorithm is a new robust clustering algorithm based on passing message on a graph, it can determine cluster number automatically and it is not affected by dimensionality. However, its time complexity is high when dealing with large and dense dataset. The most existing improvement methods process the input data matrix, instead of improving the algorithm itself. This paper improves the message passing, and clarifies that the proposed algorithm can greatly reduce the time complexity, while keeping the accuracy of original AP algorithm. The experiments illustrate that the proposed algorithm can greatly reduce the time complexity of original AP algorithm on datasets with repeated points, and perform better than the existing AP-based algorithm, equally even better than the original AP algorithm on some dataset.

Keywords: affinity propagation algorithm; large dataset; message passing

1 引言

在图像标注和视频处理等领域，聚类分析是经常被使用的图像建模方法。可获取的视频图像和图像的数量极大，在一次聚类中需要处理的特征数量相当多，且特征维数相当高。密切关系传播算法(Affinity Propagation --AP)[4]不受特征维数约束，且鲁棒性强，能自动确定类别数目，已经成功地应用于图像分类和标注[1] [12]。图像和视频带有相当大的冗余信息，在经过向量量化等维数约简之后，密集大规模数据集表现为具有重复点的数据集。而现有针对具有重复点的数据集的 AP 算法，只对输入数据集进行预处理，而没有涉及算法本身。

本论文工作获国家自然科学基金重大研究计划“视听觉信息的认知计算”资助(项目名称：基于 MDL 原理的图像语义特征分析方法研究，项目编号 90820010，项目类别：培育项目)。*通讯作者。

AP 是一种基于图的信息传播的聚类算法。原始 AP 算法是信念传播(Belief Propagation--BP)[14]和最大和算法[7]在聚类问题上的直接应用。

AP 算法避免了直接将类似样本置于同一类中的较差解决方案。AP 算法也不要求聚类问题近似超球形分布。AP 比高斯混合模型，K 均值算法，光谱聚类和层次聚类好，既能按预定的类别数目聚类，也能自动确定类别数目[3]。

密切关系传播算法除了具有较好的聚类效果外，其不要求输入全部特征值，而只要求给出特征之间的相似度矩阵。因此维数的高低，只影响到相似度矩阵的计算，而不影响 AP 算法的复杂度。图像标注问题往往需要提取各种高维特征，AP 算法这一性质对于图像标注问题非常有用。

近年来，对于 AP 算法的研究主要集中于下列问题：

1.1 参数学习问题

在一类自适应 AP 算法中[8][9]，通过调整参数以使算法尽快收敛。通过增加松弛系数到 0.85，然后减小偏好值直到算法收敛。当类别数目较高时，算法对步长比较敏感，因此步长通过类别数目的一个函数来确定。

而在另一类自适应 AP 算法中，目标在于自适应地获取最优分类数目。Frey 等人[4]指出在类别数目与 AP 算法的参数（偏好值）之间存在单调关系。为了获取期望聚类数目的聚类结果或减少迭代次数，可以通过搜索参数空间来获得合适的参数。

一些 AP 算法针对特定问题研究[12][13]。在图像标注问题中，针对特定数据集，类别数目的对数和偏好值的对数之间存在着线性关系。可以从训练集的语义信息中推断出类别数目，然后通过训练集的一个子集学习得到语义信息、类别数目和 AP 参数之间的关系。

Furtlehner 等人指出存在一个偏好值，可以区分潜在类别结构的分裂阶段和联合阶段[2]。据此，一个基于 AP 的策略被用来发现给定的数据集中的类别数目。

1.2 处理大数据集时的时间耗费问题

AP 算法基于完全图，其时间复杂度为 $O(N^2)$，因此处理大数据集时会导致缓慢的收敛速度。对于小数据集 AP 算法与 K 均值算法一样快，但对于大数据集，AP 算法慢得多[3]。研究者们发展了很多方法解决时间耗费问题。

1.2.1 处理稀疏相似度矩阵的方法

研究表明，在稀疏相似度矩阵上的时间消耗比在全连接图上的少的多[4]。为降低时间复杂度，可以人为地构造稀疏矩阵。

在原始的稀疏图上获取了代表点之后，一种分层式算法在代表点构成的一个稀疏图（两个代表点之间建立连接，如果被这两个代表点代表的至少一对点之间存在连接）上再次运行 AP 算法[11]。

1.2.2 处理具有重复点的大数据集的方法

对于具有重复点的大数据集，一个加权的 AP（Weighted AP--WAP）算法被提出[2][16]。重复点被看做一个点，其与另一个点的相似度变为乘以重复次数，其偏好度变为加上一个与重复次数有关的正数。

1.2.3 对于一般数据集的基于分而治之策略的方法

在获取整个数据集子集的代表点之后，在代表点上再次运行 AP 算法[16]。每个子集是一个全连接子图，由所有代表点构成的子图也是全连接的，但是在分而治之之后整个数据集不是全连接的。

为了处理大数据集，一种分区 AP（Partition AP--PAP）算法被提出[10]。PAP 先在数据集的子集上传递信息，然后把所有子集的可用度矩阵合并作为整个数据集的可用度矩阵。该扩展假设数据来自于一个均匀的抽样。

投票分区 AP 算法（Voted PAP--VPAP），是基于 AP 的证据累积聚类。聚类结果不在约束为超球形分布。VPAP 算法分为三步：PAP，生成松弛多根最小生成树，多数投票[17]。

Furtlehner 等人指出当维数不是很高，并且分类之间足够远时，用于大数据集的分而治之策略没有明显地损害精度[2]。

1.2.4 基于采样技术的方法

地标点技术被用于 AP 算法，在一个随机采样的小数据集上运行 AP 算法，发现代表点和他们代表的标记点[8][10]。并将剩余未标注的点分配到最近的代表点，根据的是他们到代表点的距离是否比最远的标记点近。对于仍未分配的点，迭代地运行地标点 AP 算法。

1.3 结合先验知识的问题

稀疏相似度矩阵中，缺失的边即是一种先验知识，可以理解为不存在联系（语义上“完全不同类”）。

原始 AP 算法是无监督的。为处理一些数据点已有已知标签的问题，一种半监督的 AP 算法被提出[11]。带有已知标签的点被收缩为单个点，另一个点与收缩点的相似度被设置为另一个点到所有被收缩点之间相似度的最大值。收缩点的偏好度被设置为零。

每个聚类簇只能拥有一个代表点，限制了 AP 算法的结果为规则形状的簇。为了放松这种限制，一个新的参数被引入，用来在选择最近点作为代表点的简单情况与原始 AP 之间进行插值[6]。

Yu 等人研究了 AP 算法的收敛性质，给出了当算法收敛的时候决策矩阵的性质，给出了没有松弛因子的 AP 算法准则[15]。

在一种改进的 AP 聚类算法（Improved AP--IAP）中，拥有较大网络支持似然值的点更有可能被选为聚类中心，IAP 算法被用来设置 LBG 方法的初始值[5]。

Zhu 等人提出了一种新的聚类策略，使用 AP 算法来初始化 K 均值算法[18]。

1.4 小结

如上所述，近年来的文献较深入地研究了 AP 算法的参数学习问题，但对于处理大数据集时的时间复

杂度问题，大都对输入数据集进行预处理，而没有涉及算法本身。对带有先验知识的特定问题进行建模仍需深入研究。

研究具有重复点的大数据集处理方法，对于处理密集大数据集具有启发性；经过向量量化等维数约简之后，密集大数据集可以转化为具有重复点的数据集问题。然而 WAP 方法[16]修改重复点的相似度和偏好度，在重复次数较多时，会导致重复点成为孤立聚类簇，以及出现代表点只能从重复点中产生的问题。本论文试图从消息传递机制的角度，针对具有重复点的大数据集改进 AP 算法。

2 AP 算法

依据文献[4]，我们对 AP 算法简要回顾一下：

AP 的中心思想是使用信息传播作为启发策略来寻找代表点。实值信息在数据点对之间迭代升级，直到稳定的聚类出现。

2.1 AP 的目标函数

$$S(c)=\sum_{i=1}^{N}s(i,c_i)+\sum_{k=1}^{N}\delta_k(c)，\text{其中}$$

$$\delta_k(c)=\begin{cases}-\infty & \text{如果 } c_k\neq k \text{ 且 } \exists i:c_i=k\\ 0 & \text{其他}\end{cases} \quad (1)$$

其中变量节点 $c=(c_1,c_2,...,c_N)$是数据点的类别标签。$s(i,j)$表示数据点 i 到 j 的相似度，自我相似度 $s(i,i)$即数据点 i 的偏好度。因此函数节点 $s(i,c_i)$表示数据点 i 到其代表点 c_i 的相似度。函数节点 $\delta(c)$表示惩罚项，强制代表点选择自身为代表点。

2.2 消息传递

责任度 $r(i,k)$，从样本点 i 发送给潜在的代表点 k，作为累积证据，反映了样本点 k 作为样本点的代表点 i 是否合适，相对于所有其他潜在的代表点。

可用度 $a(i,k)$，从候选代表点 k 发送到样本点 i，作为累积证据，反映了候选样本点 k 成为样本点 i 的代表点的合适程度，考虑到来自其他样本点的支持。

开始时，可用度被初始化为 0,

$$a(i,k)=0. \quad (2)$$

然后，责任度使用如下规则计算：

$$r(i,k)\leftarrow s(i,k)-\max_{k'\text{s.t.}k'\neq k}\{a(i,k')+s(i,k')\}. \quad (3)$$

用度使用如下规则计算：

$$a(i,k)\leftarrow \min\left\{0,r(k,k)+\sum_{i'\text{s.t.}i'\notin\{i,k\}}\max\{0,r(i',k)\}\right\}, \quad (4)$$

$$a(k,k)\leftarrow \sum_{i'\text{s.t.}i'\neq k}\max\{0,r(i',k)\}. \quad (5)$$

阻尼因子调节上次迭代信息保存的比率，用来防止震荡的出现。

$$R_t=(1-\lambda)R_t+\lambda R_{t-1}，\ A_t=(1-\lambda)A_t+\lambda A_{t-1} \quad (6)$$

代表点的判定：

$$e=\max_k\{a(i,k)+s(i,k)\}. \quad (7)$$

在任何时刻，可用度和责任度都可以结合起来表示一个代表点。对于样本点 i，计算它与所有样本点之间的 $a(i,j)+r(i,j)$。如果样本点 k 的值最大，那么或者 i 就是代表点（当 $k=i$ 时），或者 k 是 i 的代表点。

2.3 AP 算法的问题

1) 处理大规模数据时，时间耗费过高。

AP 算法的时间复杂度为 $O(T|\mathcal{E}|)$，其中 T 是迭代的次数，$|\mathcal{E}|$是边数[1]。对于有向完全连接图来说，$|\mathcal{E}|=N(N-1)$。随着顶点个数的增加，AP 算法的时间耗费增长较快。

2) 无法处理数据相似度矩阵之间的冗余信息。

对于大规模数据集，例如图像像素聚类，其相似度矩阵之间存在着较大的冗余和重复。原始的 AP 算法没有提供合理的机制来去除这些冗余信息。

3 面向具有重复点大数据集的 AP 算法改进

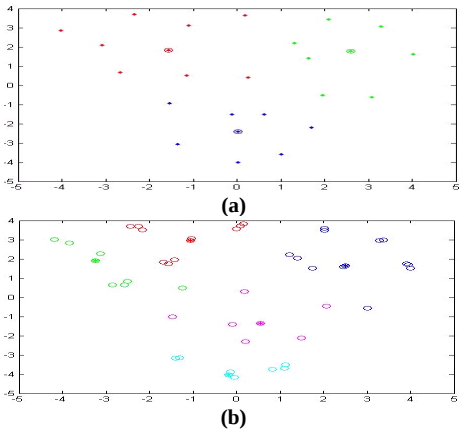

图 1 原始 AP 算法对(a)无重复点数据集和(b)重复点数据集(增加扰动以方便显示)的聚类结果对比

受到一种处理具有重复点的大数据集的 AP 算法改进 (WAP)[16] 的启发，我们提出了一种针对具有重复点的大数据集的改进的 AP 算法，称该算法为 APRP (AP Repeated Points)。WAP 改变作为算法输入的相似度矩阵，而我们试图通过改变消息处理来进行改进。

依据文献[16]，具有重复点的数据集通常表示为(x_i,n_i)，数据点 x_i 及其重复次数 n_i。对于密集大数据集，可通过向量量化构造具有重复点的数据集。

3.1 重复点及消息传递的算法分析

通过分析，我们给出如下命题：

1) 无论重复与否，两点之间的相似度只与两点的特征值有关，而与重复点无关。

2) 重复次数越多，该点成为代表点的可能性越大，所以其偏好度应越大。

3）责任度的计算与重复次数无关。考虑公式(3)中责任度 $r(i,k)$的计算，根据 $\max\limits_{k'\text{s.t.}\,k'\neq k}\{a(i,k')+s(i,k')\}$选择数据点 $x_{k'}$。具有最大值的一个点被选中，其与重复次数无关。

4) 可用度的计算与重复次数有关。考虑公式(4)(5)中可用度 $a(i,k)$和自我可用度 $a(k,k)$的计算，$\sum\limits_{i'\text{s.t.}\,i'\notin\{i,k\}}\max\{0,r(i',k)\}$是求和项，需要考虑重复次数。

3.2 原始 AP 算法的分析

我们对原始 AP(Original AP--OAP)算法进行了分析。图 1(a)是无重复点数据集，OAP 将其分为 3 类。图 1(b)是从图 1(a)上生成的具有重复点的数据集，OAP 将其分为 5 类。因此，直接把重复点看作一个点应用 OAP 算法，在处理具有重复点的数据集的聚类问题时，得不到令人满意的结果。需要对数据或者算法本身进行改进。

3.3 改变输入数据的分析

现有算法大都对输入数据集进行处理，我们对其进行了分析。以 WAP[16]为例，相似度和偏好度被改为：

$$s'(i,j)=n_i s(i,j), \tag{8}$$

$$s'(i,i)=s(i,i)+(n_i-1)\varepsilon_i,\ \varepsilon_i\geq 0, \tag{9}$$

其中 n_i 是重复次数。公式(9)符合 3.1 节的第 2 条，而公式(8)不符合 3.1 节第 1 条。

图 3,4,5 显示了改变相似度和改变责任度对于 AP 算法的作用，图 3 只根据公式(8)改变相似度，图 4 只根据公式(9)改变偏好度，图 5 是同时应用公式(8)和公式(9)的结果。

公式(8)的负面作用是：在重复次数较大的情况下，公式(8)会导致形成过多的孤立聚类簇。如图 2 所示，重复 2 次的点 C 的实际位置位于虚线处。按公式(8)，C 与其他点的相似度被乘以 2。由于在 AP 算法中相似度定义为负值，所以 C 与其他点之间的相似程度降低（直观地表现为距离拉远）。一个极端的例子是当 n_i 足够大，重复点 i 将会距离其他点足够远因此重复点只能形成一个单独的聚类簇。而按照重复点的实际位置，重复点本应“吸引”周围的非重复点点形成聚类簇（图 3 的实验证明了这一点）。

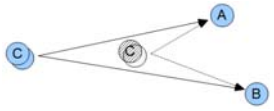

图 2. 改变重复点相似度的示意图

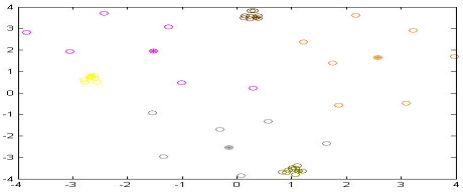

图 3 只改变相似度，重复点形成了三个孤立聚类簇

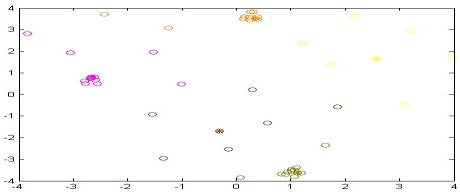

图 4 只改变偏好度，重复点倾向于成为聚类中心

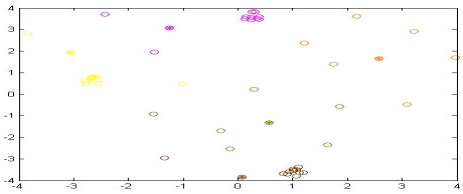

图 5 同时改变相似度和偏好度，聚类结果与图 4 类似

公式(9)的正面作用是：改变偏好度能使重复点吸引更多的点形成聚类簇。这在重复次数不大时是有效的（图 4）。其负面作用是：在重复次数较大时，反映成为代表点可能性的偏好度也会较大，公式(9)会使代表点只能从重复点中产生。看起来 WAP （同时改变相似度和偏好度）算法（图 5）中改变偏好度引起的正面作用超过了改变相似度引起的负面作用，因此图 5 的分类结果与图 4 类似。

如上所述，WAP 算法中，改变输入数据相似度的作法是不成功的，算法上背离了重复点的定义。

3.4 改变消息传递

本文算法 APRP 把优化问题的目标函数修改为：

$$S(c)=\sum_{i=1}^{N} n(i)s(i,c_i)+\sum_{k=1}^{N}\delta_k(c). \quad (10)$$

依据 3.1 节第 1 条，APRP 算法不改变相似度 $s(i,j)$。依据 3.1 节第 2 条，偏好度按 WAP[16]方式改变：

$$s'(i,i)=s(i,i)+(n_i-1)\varepsilon_i,\ \varepsilon_i\geq 0. \quad (11)$$

依据 3.1 节第 3 第 4 条，不改变责任度公式，可用度公式变为：

$$a(i,k)\leftarrow \min\left\{0, r(k,k)+\sum_{i'\text{ s.t. }i'\notin\{i,k\}}\max\{0,n(i')r(i',k)\}\right\}, \quad (12)$$

$$a(k,k)\leftarrow \sum_{i'\text{ s.t. }i'\notin\{i,k\}}\max\{0,n(i')r(i',k)\}. \quad (13)$$

4 实验研究

4.1 实验数据及评价准则

实验将对下列方法进行对比分析：OAP[4], WAP[16], APRP。

4.1.1 无重复点的数据集

实验中采用了两个数据集[19]：Toy Problem (DB1), Travel Routing (DB2)，两个数据库都提供了相似度矩阵和偏好度矩阵，只有 DB1 提供了数据特征值。图 1,3,4,5 使用了来自 DB1 的二维数据。我们使用来自 DB1, DB2 数据生成重复点数据集来验证 APRP。

4.1.2 从普通数据集生成重复数据

生成带有重复点的数据集的方法如下，主要采用公式

$$(x_i,n_i),\ n_i=R_j,\ j=\text{ceil}(M\cdot\gamma_i). \quad (14)$$

其中 n_i 是数据点 x_i 的重复次数，γ_i 是[0,1]之间的一个随机浮点数。$[R_1,...,R_M]$是用户定义的重复“种子”，从中每个数据点 x_i 随机选择一个重复次数 n_i。图 1(b)的重复数据是由“种子”[1, 2, 3]生成的。

对于 DB1 和 DB2 分别产生了 11 个具有重复点的数据集，其$[R_1,...,R_M]$如下：113, 123, 124, 125, 133, 135, 333, 1x9_10, 1x9_20, 1x7_4_7_10。其中 1x9_10 表示包含 9 个 1 和 1 个 10。

4.1.3 聚类结果评价准则

本文采用文献[20]中的评价准则衡量聚类结果的优劣：

$$f=\sum_{i=1}^{N}|x_i-s_i|+\sum_{i=1}^{S}|s_i-\bar{s}|, \quad (15)$$

其中，$\sum_{i=1}^{N}|x_i-s_i|$ 表示类内到类心距离之和；$\sum_{i=1}^{S}|s_i-\bar{s}|$ 表示代表点到其均值的距离之和，用来抑制过多的类别数目。

由于实验数据集 DB2 没有提供数据特征值，各代表点 s_i 到其均值 $\bar{s}$ 的距离无法直接求得。我们使用如下计算规则选择距离均值 $\bar{s}$ 最近的代表点 e：

$$e=\arg\max_k\left\{\sum_{j\neq k}^{S}s(k,j)\right\}. \quad (16)$$

4.2 具有重复点的数据集的实验结果

表 1：DB1 的类别数和聚类结果

	类别数			聚类结果		
	OAP	WAP	APRP	OAP	WAP	APRP
113	5	4	5	64	70	60
123	5	5	5	69	74	70
124	6	7	12*	74	85	73*
125	8	8	7	79	95	88
133	6	8	5	78	90	85
135	6	8	8	84	111	91
333	8	13*	9	86	78*	87
1x9_10	5	5	5	47	81	47
1x9_20	4	3	4	45	89	45
1x7_4_7_10	6	7	9	63	90	80

表 2：DB2 的类别数和聚类结果

	类别数			聚类结果		
	OAP	WAP	APRP	OAP	WAP	APRP
113	11	11	14	7860	8040	7390
123	1*	12	13	216*	9500	8710
124	14	12	15	9382	11328	9321
125	14	11	17	11489	13641	10732
133	14	10	14	9830	12452	9812
135	15	13	18	12715	14808	12242
333	15	29*	17	12227	10306*	11881
1x9_10	11	53	16	8501	4649*	7930
1x9_20	12	41*	38*	10303	4866*	4709*
1x7_4_7_10	14	80	22	12097	81832*	10733

按公式(15)衡量数据到聚类中心的距离之和，比较每一行数据集上三个算法的聚类结果，带下划线的数据表示其中的最好结果。星号表示 WAP 或 APRP 的异常实验结果，其异常现象为类别数远大于 OAP 的结果。

表 3：DB2 的时间耗费

Time(s)	OAP	WAP	APRP
113	31	8	9
123	42	8	9
124	52	8	9
125	75	8	9
133	57	8	9
135	101	8	9
333	95	8	9
1x9_10	38	8	9
1x9_20	61	8	9
1x7_4_7_10	87	8	9

从表 1 的聚类结果可以看出，DB1 生成的 11 个数据集中，在 10 个数据集上是 OAP 结果最好，在 3 个数据集上 APRP 结果最好（2 个数据集上二者并列第一）。总体上 APRP 比较接近于 OAP 的结果，而 WAP 则结果较差。从表 2 的聚类结果可以看出，DB2 生成的 11 个数据集上，在 10 个数据集上 APRP 结果最好，在一个数据集上 OAP 结果最好。总体上 OAP 比较接近于 APRP 的结果，而 WAP 算法则结果较差。

AP 算法的时间复杂度为 $O(N^2)$。例如对数据集 DB1-113 来说，OAP 算法处理的点大约是 WAP 和 APRP 处理的点的(1+1+3)/3 倍。表 3 对比了三种算法的时间耗费，可以看出原始 OAP 算法耗费的时间数倍于其余两种算法。因此，综合考虑时间复杂度与精度，APRP 是三者中最优的对具有重复点的大数据集的算法。

5 总结

在使用 AP 算法处理具有重复点的大数据集时，对算法中的偏好度设置和可用度消息传递部分做了相应的改进，在保持 AP 算法的鲁棒性和自动确定分类数目等优点的同时，大幅降低了 OAP 算法的时间耗费。本文通过分析得出了处理具有重复点数据集时 AP 算法应该满足的四个命题，并通过修改消息传递机制对 OAP 进行了改进。相比于 WAP 等处理具有重复点数据集的算法，APRP 方法在精度上最接近（在部分数据集上甚至略优于）OAP 算法，在时间耗费和精度上达到了一个很好的平衡。若结合向量量化技术[21]，该算法可高效地应用于视频处理、图像标注等具有密集冗余信息的大数据集问题。

参考文献

[1] D. Dueck and B.J. Frey, “Non-metric affinity propagation for unsupervised image categorization,” *In: IEEE International Conf. on Computer Vision*, pp. 1-8, 2007.

[2] C. Furtlehner, M. Sebag and X.L. Zhang, “Scaling analysis of affinity propagation,” *Physical Review E*, vol. 81, no. 6, pp. 006102, 2010.

[3] B.J. Frey and D. Dueck, “Mixture modeling by affinity propagation,” *In: Advances in Neural Information processing Systems*, pp. 379-386, 2006.

[4] B.J. Frey and D. Dueck, “Clustering by passing messages between data points,” *Science*, vol. 315, pp. 972-976, 2007.

[5] W. Jiang, F. Ding and Q.L. Xiang, “An affinity propagation based method for vector quantization codebook design,” *In: International Conf. on Pattern Recognition*, pp. 1-4, 2008.

[6] M. Leone, Sumedha and M. Weigt, “Clustering by soft-constraint affinity propagation: applications to gene-expression data,” *Bioinformatics*, vol. 23, no. 20, 2708-2715, 2007.

[7] F.R. Kschischang, B.J. Frey and H.A. Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498-519 , 2001.

[8] C.Y. Sun, C.H. Wang, S. Song and Y.-F. Wang, “A local approach of adaptive affinity propagation clustering for large scale data,” *In: International Joint Conf. on Neural Networks*, pp. 161-165, 2009.

[9] K.J. Wang, J.Y. Zhang, D. Li, X.N. Zhang and T. Guo, “Adaptive affinity propagation clustering,” *Acta Automatica Sinica*, vol. 33, no.12, pp. 1242-1246, 2007.

[10] D.Y. Xia, F. Wu, X.Q. Zhang and Y.T. Zhuang, “Local and global approaches of affinity propagation clustering for large scale data,” *Journal of Zhejiang University SCIENCE A*, vol. 9, no. 10, pp. 1373-1381, 2008.

[11] J.X. Xiao, J.D. Wang, P. Tan and L. Quan, “Joint affinity propagation for multiple view segmentation,” *In: International Conf. on Computer Vision*, pp. 1-7, 2007.

[12] D. Yang and P. Guo, “Improvement of image modeling with affinity propagation algorithm for image semantic annotation,” *In: Proc of International Conf. on Neural Information Processing*, pp. 778-787, 2009.

[13] D. Yang and P. Guo, “Image modeling with combined optimization techniques for image semantic annotation,” *Neural Computing and Applications*， 2010 (In Press)

[14] J.S. Yedidia, W.T. Freeman and Y. Weiss, “Understanding belief propagation and its generalizations,” *Exploring artificial intelligence in the new millennium*, pp. 239-269, 2003.

[15] J. Yu and C.Y. Jia, “Convergence analysis of affinity propagation,” *In: Knowledge Science, Engineering and Management*, pp. 54-65, 2009.

[16] X.L. Zhang, C. Furtlehner and M. Sebag, “Data streaming with affinity propagation,” *In: Machine Learning and Knowledge Discovery in Databases*, pp. 628-643, 2008.

[17] X.Q. Zhang, F. Wu and Y.T. Zhuang, “Clustering by evidence accumulation on affinity propagation,” *In: International Conf. on Pattern Recognition*, pp. 1-4, 2008.

[18] Y. Zhu, J. Yu and C.Y. Jia, “Initializing k-means clustering using affinity propagation,” *In: International Conference on Hybrid Intelligent Systems*, pp. 338-343, 2009.

[19] http://www.psi.toronto.edu/affinitypropagation/webapp/

[20] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 3rd ed. Salt Lake City USA :Academic Press, Chp.11, Sec.2, 2006.

[21] S. Lin, Y. Yao and P. Guo, “Speed up Image Annotation Based on LVQ Technique with Affinity Propagation Algorithm,” accepted, *In: 17th International Conference on Neural Information Processing*, 2010.