高技术通讯 2012 年 第22 卷 第7 期:697 ~ 705

doi:10.3772/j.issn.1002-0470.2012.07.005

基于相容粒度空间模型的自适应图像语义分类方法①

蒙祖强②* 史忠植**

(*广西大学计算机与电子信息学院 南宁 530004)

(**中国科学院计算技术研究所智能信息处理重点实验室 北京 100190)

摘 要 针对图像底层特征和高层语义之间存在的语义鸿沟问题,运用相容粒度空间模型对图像语义分类进行了研究,提出一种自适应的图像语义分类方法,为解决此问题探索出了一种有效途径。该方法将图像集建模为基于原始特征的相容粒度空间;在此空间中,通过引入相容参数和构造距离函数来定义相容关系,从而通过调整相容参数可有效控制对象邻域粒的大小,最终可直接处理图像的实数型特征而无需进行离散化等预处理;此外,通过引入相容度的方法实现对相容参数的自适应优化,从而自动调整邻域粒的大小,使得构造的分类器几乎不需要手工设置参数即可自动适应于各种不同类型的图像集,并获得比同类算法更好的分类准确率。实验结果验证了这种方法的有效性和可行性。

关键词 图像语义分类,粒度计算(GrC),自适应,相容粒度空间,相容关系

0 引 言

图像语义分类是将图像标记为不同语义类别的过程。由于高层语义与低层视觉特征之间存在语义鸿沟[1],视觉特征相似的图像在语义上可能是不相关的,因此如何实现对大量图像的自动语义分类和标注,则成为目前极具挑战性的研究课题[2,3]。

由于存在语义鸿沟,需要在低层视觉特征与高层语义之间构建有效的映射,以实现自动的图像语义分类和语义检索,使用的主要方法和技术是Bayes方法、神经网络、支持向量机(SVM)、决策树技术等。Vailaya 等[4]较早将 Bayes 方法用于图像语义检索和分类研究,主要用 Bayes 公式实现了底层特征到高层语义映射的层次描述。之后,他们进一步研究了基于 Bayes 的图像语义分类,主要讨论休假图像的层次分类问题[5]。文献[6]利用 Bayes 统计学习和决策理论,建立了一种基于描述特征建模方法的图像语义综合概率描述模型,由此实现了图像的语义分类和检索。Bayes 方法通常要求属性之间是相互独立的,这在实际应用中往往难以满足,特别是在维数较多的情况下尤其如此。由于图像数据的复杂性和高维性,神经网络以其良好的"黑箱"学习能力在图像分类中得到广泛应用[7,8],但其有自身的不足:构造的神经网络往往具有比较复杂的结构[8],训练速度慢,且一般没有良好的数学基础,难以对其进行理论分析,所形成的方法通常针对特定的应用案例而不具普遍意义。另一类广泛使用的图像分类技术是 SVM,它具有很强的理论基础[9-12],但是其本质上是二类分类器,在解决多分类问题时需要进行结构性改变,随着语义类别的增加,其效率和准确率会大受影响。有的方法则通过构建中间语义层来架起底层特征和高层语义之间的"桥梁"。例如,文献[2]通过引入主题语义模型,在底层特征和高层语义之间构建隐含概率语义模型,但主题模型缺乏对不同区域间内容关联和空间关系的充分考虑。文献[13]虽然扩展了主体语义模型,考虑了不同区域之间的关系,但该方法在统计特征信息时受制于预先提供的经验知识,其实需要大量的参数设置,其分类效果在很大程度上依赖于领域知识专家。

近年来,粒度计算(granular computing, GrC)在图像处理方面逐步得到应用和推广,并显示出较好的效果[14]。我们课题组也一直从事有关粒度计算在图像处理方面的理论和应用研究,初步形成了相

① 863 计划(2007AA01Z132),973 计划(2007CB311004),国家自然科学基金(61063032)和广西自然科学基金(2012GXNSFAA053225)资助项目。

② 男,1974 年生,博士,教授;研究方向:图像理解,粒度计算,机器学习等;联系人,E-mail: zqmeng@126.com

(收稿日期:2011-05-25)

容粒度空间模型[15-17]。在我们的模型中,粒(granules)是允许相交的,而在商空间理论、Rough 集等粒度计算模型中,粒通常是不允许相交的,这在许多情况(特别是处理原始图像特征)下难以满足。因此,相容粒度空间模型在实际应用中具有更好的普适性。本文进一步对相容粒度空间模型进行扩充,并在参考相容粒度划分方法[18]的基础上提出了一种基于相容粒度空间模型的自适应图像语义分类算法。与已有算法相比,该算法可以直接处理连续型数值特征,在一定程度上可避免因特征离散化而造成的有用信息的丢失,几乎不需要手工对参数进行选择和设置,可以自动适应于不同应用背景下的图像语义分类,并能获得比其他同类算法更好的分类性能。

1 基于图像特征的相容粒度空间

1.1 相容粒度空间的概念

令 U 表示图像向量特征的集合,简称图像集;令 $F=\{a_1, a_2, \cdots, a_n\}$ 表示图像特征的集合,V_a 表示特征 a 的值域。图像特征值通常是实数型,因此 V_a 为一有限的实数集。图像 x 在特征 a 上的取值表示为 $f(x, a)$,即 f 是 U 到 V_a 的映射。对于特征 a,构造一函数 $dis: U\times U\to R^*: dis(x, y; a)=|f(x, a)-f(y, a)|$,其中 $x, y\in U$,R^* 表示非负实数集。显然,$dis(x, y; a)$ 是 $X\subseteq U$ 上的距离函数,因此 (X, dis) 为关于特征 a 的距离空间。令 $Tol(X, a, \lambda)=\{(x, y)\mid dis(x, y; a)\leqslant\lambda, x, y\in X\}$,则 $Tol(X, a, \lambda)$ 满足自反性和对称性,即 $Tol(X, a, \lambda)$ 是 X 上关于 a 的一个相容关系。

定义 1 二元组 $(X, Tol(X, a, \lambda))$ 称为关于 a 的相容空间,λ 称为 a 的相容参数。在相容空间 $(X, Tol(X, a, \lambda))$ 中,由相容关系 $Tol(X, a, \lambda)$ 对 X 划分而形成的(最大)相容类被视为一种彼此不可区分的对象的集合,称为关于 a 的(最大)相容粒(tolerance granule)。

定义 2 令 $S_{Tol(X, a, \lambda)}(X, x)=\{y\mid (x, y)\in Tol(X, a, \lambda), y\in X\}, x\in X$,则 $S_{Tol(X, a, \lambda)}(X, x)$ 称为 X 上 x 关于特征 a 的邻域,简称 x 的邻域(有的文献称为邻域粒)。

定义 3 $TGS(X, a, \lambda)$ 称为 X 上关于特征 a 的相容粒度空间(tolerance granular space),如果 $TGS(X, a, \lambda)$ 是表示满足下列条件的若干相容粒的集合:$\cup TGS(X, a, \lambda)=X; \varnothing\notin TGS(X, a, \lambda)$,其中 $\cup$ 表示集合的广义并。

下面给出计算 X 上关于特征 a 的相容粒度空间 $TGS(X, a, \lambda)$ 的算法。

算法 1 计算 $TGS(X, a, \lambda)$, $X\subseteq U$.

输入:X, a, λ.

输出:$TGS(X, a, \lambda)$.

(1) 对 $V_a=\{f(x_1, a), f(x_2, a), \cdots, f(x_{|X|}, a)\}$ 进行升序排列;

(2) 令 $curp=1$, $i=1$, $tmp=\varnothing$;

(3) 令 $fg=0$;

(4) 如果 $(curp<|V_a|)$ 且 $|V_a[i]-V_a[curp]|<=\lambda$,则转(5),否则转(6);

(5) $curp++$, $fg=1$, 转(4);

(6) 如果 $fg=1$ 或者 $curp=1$,则将 V_a 中的第 i 个至第 $curp$ 个元素组成一个相容粒并加入到 tmp;

(7) $i++$;

(8) 如果 $i<|V_a|$,则转(3),否则转(9);

(9) 令 $TGS(X, a, \lambda)=tmp$.

由步骤(6)知道,该算法产生的相容粒都是最大相容粒,因而获得的是最大相容粒度空间。算法 1 中控制变量 i 和 $curp$ 都没有回溯,(2)至(9)的运行时间接近于 $O(|X|)$,故该算法的复杂度主要由(1)确定,一般为 $O(|X|\log(|X|))$。

定义 4 对给定的两个相容粒度空间 $TGS(X, a_1, \lambda_1)$ 和 $TGS(X, a_2, \lambda_2)$,定义相容粒度空间之间的运算 $\cap^*$ 如下:$TGS(X, a_1, \lambda_1)\cap^* TGS(X, a_2, \lambda_2)=\{Z\in TGS(Y, a_2, \lambda_2)\mid Y\in TGS(X, a_1, \lambda_1)\}$。

为表述方便,将 $TGS(X, a_1, \lambda_1)\cap^* TGS(X, a_2, \lambda_2)$ 的结果表示为 $TGS(X, a_1, \lambda_1; a_2, \lambda_2)$。此外,运算 $\cap^*$ 不满足交换律,因此 $TGS(X, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_n, \lambda_n)$ 中的参数是有先后顺序的。

利用算法 1,进一步设计计算相容粒度空间 $TGS(U, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_m, \lambda_m)$ 的算法:

算法 2 计算相容粒度空间 $TGS(U, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_m, \lambda_m)$, $m\leqslant n$, n 表示特征数.

输入:$U, a_1, a_2, \cdots, a_m, \lambda_1, \lambda_2, \cdots, \lambda_m$.

输出:$TGS(U, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_m, \lambda_m)$.

(1) 令 $tmp1=TGS(U, a_1, \lambda_1)$; //利用算法 1

(2) for $i=2$ to m do

(3) {令 $tmp2=tmp1$, $tmp1=\varnothing$;

(4) for each $X\in tmp2$ do $tmp1=tmp1\cup TGS(X, a_i, \lambda_i)$;

//利用算法 1

}

(5) 令 $TGS(U, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_m, \lambda_m) = tmp1$.

该算法的计算时间主要由(2)至(4)决定,其复杂度接近于 $O(m \mid U \mid \log(\mid U \mid))$。

1.2 相容粒度空间的性质

定理 1 令 $T(X, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_m, \lambda_m) = Tol(X, a_1, \lambda_1) \cap Tol(X, a_2, \lambda_2) \cap \cdots \cap Tol(X, a_m, \lambda_m)$, $X \subseteq U$, $m \leqslant n$, 则 $T(X, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_m, \lambda_m)$ 为相容关系。

显然,此定理无需证明,因为有限个相容关系的交集仍然是相容关系。

相容关系 $T(X, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_m, \lambda_m)$ 主要是由特征集 $\{a_1, a_2, \cdots, a_m\}$ 及相容参数 $\lambda_1, \lambda_2, \cdots, \lambda_m$ 决定,而每个特征 a_i 都有相应的相容参数 λ_i 与之对应。为讨论方便,省略相容参数,即 $T(X, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_m, \lambda_m)$ 简写成 $T(X, \{a_1, a_2, \cdots, a_m\})$;如果 $X = U$, 则 $T(X, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_m, \lambda_m)$ 可以进一步简写成 $T(\{a_1, a_2, \cdots, a_m\})$。

注意,相容关系 $T(X, a_1, \lambda_1; a_2, \lambda_2; \cdots; a_m, \lambda_m)$ 与特征 $a_1, a_2, \cdots, a_m$ 的先后顺序是无关的。

定义 5 令 $S_{T(F')}(x) = \{y \mid (x, y) \in T(F'), y \in U\}$, $x \in U$, 则 $S_{T(F')}(x)$ 称为 x 关于特征集 F' 的邻域(粒)。

定理 2 令 $j_1, j_2, \cdots, j_m$ 为 $1, 2, \cdots, m$ 的一个置换,令 $F' = \{a_1, a_2, \cdots, a_m\}$, $F'' = \{a_{j1}, a_{j2}, \cdots, a_{jm}\}$, 则 $S_{T(F')}(x) = S_{T(F'')}(x)$。

限于篇幅,该定理的证明省略。定理 2 说明:(1) 不管在什么样的相容粒度空间中,只要它们的特征集及其对应的相容参数分别相同,任意 x 在这些空间中的邻域 $S_{T(F')}(x)$ 都是一样的,这是相容粒度空间的不变性;(2) 定理 2 实际上是给出了计算 $S_{T(F')}(x)$ 的程序实现方法。

定理 3 如果 $F' \subseteq F''$, 则 $S_{T(F'')}(x) \subseteq (S_{T(F')}(x)$。

同样,该定理无需证明。它表明, $S_{T(F')}(x)$ 随着特征个数的减少而"扩大"。

2 基于相容粒度空间的特征选择

2.1 相关概念及性质

对于图像 $x \in U$, 令函数 $g(x)$ 表示图像 x 的语义类标识,用 $g^{-1}(y)$ 表示语义类标识为 y 的图像的集合。这样, $g^{-1}(g(x))$ 则表示所有与图像 x 有相同语义类标识的图像的集合。

定义 6 对于特征 $a \in F' \subseteq F$, 如果 $S_{T(F'-\{a\})}(x) \subseteq g^{-1}(g(x))$, 则 a 称为图像 x 的当前冗余特征,否则 a 称为图像 x 的当前必要特征。用 $core(F', x)$ 表示图像 x 的当前必要特征的集合,即 $core(F', x) = \{a \in F' \mid S_{T(F'-\{a\})}(x) \not\subseteq g^{-1}(g(x))\}$。

定理 4 假设 b 为图像 x 的当前冗余特征,则 $core(F', x) \subseteq core(F' - \{b\}, x)$。

根据定理 3 即可证明此定理。该定理说明,如果一个特征是当前必要的,则在此后的删除步骤中也是当前必要的,而不会变成冗余特征。

定义 7 对于图像集 U, 如果特征 $a \in F' \subseteq F$ 满足条件: $\forall x \in U$, $S_{T(F'-\{a\})}(x) \subseteq g^{-1}(g(x))$, 即对所有 $x \in U$, a 均为 x 的当前冗余特征,则 a 称为 U 的当前冗余特征,否则 a 称为 U 的当前必要特征。

显然,如果 a 为 U 的当前冗余特征,则可以直接将之删除。用 $core(F', U)$ 表示 U 的当前必要特征的集合,则可以得到与定理 4 类似的结论——定理 5。

定理 5 假设 b 为 U 的当前冗余特征,则 $core(F', U) \subseteq core(F' - \{b\}, U)$, $F' \subseteq F$。

2.2 图像特征的依赖性

如上所述,不同的特征检查和删除顺序将得到不同的相容粒度空间,从而可能获得不同的必要特征。显然,我们希望先删除那些"不重要的"特征,以使最终能够选择到"重要的"特征。这可以通过分析特征的依赖性来解决。

定义 8 令 $\rho_{T(F')}(U) = \{x \mid S_{T(F')}(x) \subseteq g^{-1}(g(x)), x \in U\}$, $\rho_{T(F')}(U)$ 称为关于 F' 的正区域,其中 $F' \subseteq F$。

定义 9 令 $\gamma_{T(F')}(U) = \mid \rho_{T(F')}(U) \mid / \mid U \mid$, $\gamma_{T(F')}(U)$ 称为语义类对特征集 F' 的依赖程度,其中 $F' \subseteq F$; 又令 $\mathrm{sig}_{T(F)}(U, a) = \gamma_{T(F)}(U) - \gamma_{T(F-\{a\})}(U)$, 则 $\mathrm{sig}_{T(F)}(U, a)$ 表示特征 a 的重要程度,其中 $a \in F$。

我们要找的是语义类与图像特征之间的内在关联,因此总是希望找到"最简洁"的特征集 F', 使得图像的语义类对特征集 F' 的依赖程度最大。特征的重要程度可以在一定程度上帮助我们解决这个问题:计算所有特征的重要程度,然后按重要程度对特征进行升序排列;在检查和删除时,从左到右(重要程度从小到大)依次对它们进行检查和删除操作,直到检查完升序队列中所有的特征为止。这个过程只需要对特征集进行一遍扫描,在经过删除操作后,剩下的都是必要特征(根据定理 5)。

2.3 图像特征选择算法

根据定理 2,下面先给出基于相容粒度空间 $TGS(U,a_1,\lambda_1;a_2,\lambda_2;\cdots;a_m,\lambda_m)$ 对所有 $x\in U$ 计算 $S_{T(F')}(x)$ 的算法,然后给出计算 $sig_{T(F)}(U,a)$ 的算法,最后给出特征选择算法。

算法 3 对所有 $x\in U$,计算 $S_{T(F')}(x)$.

输入:$TGS(U,a_1,\lambda_1;a_2,\lambda_2;\cdots;a_m,\lambda_m)$.

输出:$S_{T(F')}(x),x\in U,F=\{a_1,a_2,\cdots,a_m\}$.

(1) 对所有 $x\in U$,令 $S_{T(F')}(x)=\varnothing$;

(2) for each $X\in TGS(U,a_1,\lambda_1;a_2,\lambda_2;\cdots;a_m,\lambda_m)$ do

(3) { for each $x\in X$ do

(4) { $S_{T(F')}(x)=S_{T(F')}(x)\cup X$; }

(5) }.

该算法是对 $TGS(U,a_1,\lambda_1;a_2,\lambda_2;\cdots;a_m,\lambda_m)$ 中的各个相容粒所包含的对象进行一遍扫描。所涉及对象总数略大于 $|U|$(因为有部分是重复的,具体情况与 λ_i 的大小有关),故算法的复杂度约为 $O(|U|)$。

算法 4 对所有 $a\in U$,计算 $\mathrm{sig}_{T(F)}(U,a)$.

输入:$U,a_1,a_2,\cdots,a_n,\lambda_1,\lambda_2,\cdots,\lambda_n$.

输出:$\mathrm{sig}_{T(F)}(U,a),a\in F=\{a_1,a_2,\cdots,a_n\}$.

(1) 计算 $TGS(U,a_1,\lambda_1;a_2,\lambda_2;\cdots;a_n,\lambda_n)$;
//利用算法 2

(2) 计算 $S_{T(F)}(x),x\in U$; //利用算法 3

(3) 令 $sum1=0$; //用于计算 $|\rho_{T(F)}(U)|$

(4) for each $x\in U$ do if $S_{T(F)}(x)\subseteq g^{-1}(g(x))$ then $sum1++$;

(5) for ($i=1;i<=n;i++$)

(6) { 计算 $TGS(U,a_1,\lambda_1;\cdots;a_{i-1},\lambda_{i-1};a_{i+1},\lambda_{i+1}\cdots;a_n,\lambda_n)$; //利用算法 2

(7) 计算 $S_{T(F-\{ai\})}(x),x\in U$; //利用算法 3

(8) 令 $sum2=0$;

(9) for each $x\in U$ do if $S_{T(F-\{ai\})}(x)(g^{-1}(g(x))$ then $sum2++$; //计算 $|\rho_{T-\{ai\}}(U)|$

(10) $\mathrm{sig}_{T(F)}(U,a_i)=(sum1-sum2)/|U|$; //计算 $\mathrm{sig}_T(U,a_i)$

(11) }.

该算法耗时的步骤是(6),其复杂度为 $O(n(n-1)|U|\log(|U|))$,其中 n 为特征数。利用前面设计的算法,下面给出特征选择算法。

算法 5 特征选择.

输入:$U,F=\{a_1,a_2,\cdots,a_n\},\lambda_1,\lambda_2,\cdots,\lambda_n$.

输出:特征集 F'.

(1) 对所有 $a\in F$,计算 $\mathrm{sig}_{T(F)}(U,a)$; //利用算法 4

(2) 按 $\mathrm{sig}_{T(F)}(U,a)$ 值大小,对 $a_1,a_2,\cdots,a_n$ 进行升序排列,假设结果为 $F'=<a'_1,a'_2,\cdots,a'_n>$;

(3) for $i=1$ to n do

(4) { 计算 $TGS(U,a_1,\lambda_1;\cdots;a_{i-1},\lambda_{i-1};a_{i+1},\lambda_{i+1}\cdots;a_n,\lambda_n)$;

(5) 计算 $S_{T(F-\{ai\})}(x),x\in U$;

(6) 令 $fg=1$;

(7) for each $x\in U$ do if $S_{T(F-\{ai\})}(x)\not\subseteq g^{-1}(g(x))$ then {$fg=0$; break;};

(8) if $fg=1$ then 令 $F'=F'-\{a_i\}$;

(9) }.

显然,此算法的耗时步骤为(1)和(4),它们的计算时间都为 $O(n(n-1)|U|\log(|U|))$。因此,此算法的复杂度亦为 $O(n(n-1)|U|\log(|U|))$。

3 面向图像语义分类的自适应分类器及其简化

从图像特征空间看,经过特征选择以后,冗余的列均已经被删除。但要构造一个具有高泛化能力的分类器,还有 3 个问题需要解决:(1)针对每个特征向量进一步做特征选择;(2)去掉“功能”重复的规则;(3)相容参数针对不同数据集的自适应优化,以构造自适应分类器。本节对此进行讨论。

假设经过特征选择后,获得的特征集为 $F'=\{a_1,a_2,\cdots,a_{n'}\}$,而且这些特征已经按照特征重要度进行了升序排列,其对应的相容参数分别为 $\lambda_1,\cdots,\lambda_{n'}$。

3.1 分类器的构造

本文涉及的分类器是由决策规则组成。算法 6 是在特征选择的基础上用于构造此分类器的算法。其基本思想是,依次对每个对象进行特征值筛选。

算法 6 构造分类器.

输入:$U,F'=\{a_1,a_2,\cdots,a_{n'}\},\lambda_1,\lambda_2,\cdots,\lambda_{n'}$.

输出:规则集 U'(分类器).

(1) for each $x\in U$ do

(2) { 令 $F''=F'$;

(3) for $i=1$ to n' do

(4) { 计算 $S_{T(F''-\{ai\})}(x)$;

(5) if $S_{T(F'-\{ai\})}(x)(g^{-1}(g(x))$ then $F''=F''-\{a_i\}$; }

(6) for $i=1$ to n' do if $a_i\notin F''$ then $f(x,a_i)=-1$;
//-1 表示对应的特征值被忽略,相当于被删除

(7) }.

对给定的 x,计算 $S_{T(F''-\{a_i\})}(x)$ 的最坏时间约为 $O(|F''||U|)$,因此算法的最坏复杂度为 $O(|F''|^2|U|^2)$。实际上,由于 $|F''|$ 通常远小于 $n=|F|$,并且很多时候算法并没有达到最坏复杂度的情况。因此,相对于算法 5 而言,算法 6 的运行过程是"瞬时的"。

算法 6 中,对于 U' 中被删除的那些特征值用 -1 来填补(-1 表示相应的特征值无意义)。这样,U' 中规则的长度与样本集 U 中样本的长度是相等的。

3.2 分类器的简化

这里要涉及到样本与规则的匹配问题。下面先引入关于匹配的几个概念,然后再给出简化算法。

定义 10 对于规则 $r\in U'$ 以及样本 $x\in U$,对所有 $f(r,a_i)$ 不等于 -1 的 a_i,如果 $|f(x,a_i)-f(r,a_i)|\leqslant\lambda_i$,则称样本 x 与规则 r 相容匹配,$i\in\{1,2,\cdots,n\}$。

定义 11 对于规则 $r\in U'$ 以及样本 $x\in U$,定义函数 $d(r,x)=\sqrt{\sum_{i\in A}(f(r,a_i)-f(x,a_i))^2}/|A|$,其中 $A=\{i\in\{1,2,\cdots,n\}\mid f(r,a_i)\neq-1\}$,则 $d(r,x)$ 称为 r 与 x 的平均欧式距离。

定义 12 在所有与规则 r 相容匹配的样本中,如果样本 x 与规则 r 的平均欧式距离 $d(r,x)$ 最小,则称样本 x 与规则 r 基于最小平均欧式距离的相容匹配,简称匹配。

定义 13 如果样本 x 与规则 r 匹配,则称 r 覆盖 x,或称 x 被 r 覆盖。用 $\Re(x)$ 表示所有覆盖样本 x 的规则的集合;用 $\mathscr{D}(r)$ 表示所有被规则 r 覆盖的样本的集合。

对分类器进行简化的基本思想是:如果 $\mathscr{D}(r_1)\subseteq\mathscr{D}(r_2)$,则规则 r_1 应该予以删除;在同等条件下,$|\mathscr{D}(r)|$ 值小的规则应该优先被删除。注意,直接按照此思路去实现是有困难的。但我们通过逐一地对每一个语义类的样本进行处理,巧妙地实现了删除不必要的规则、达到简化分类器的目的。

据此考虑,下面给出用于简化分类器的算法。

算法 7 分类器简化算法.

输入:图像集 U 和规则集 U',含特征和相容参数信息.

输出:简化的规则集 U''(分类器).

(1) $U''=\varnothing$;

(2) 对所有 $r_i\in U'$,计算 $|\mathscr{D}(r_i)|$,$i=1,2,\cdots,s$,并按 $|\mathscr{D}(r_i)|$ 对 $\{r_1,r_2,\cdots,r_s\}$ 进行升序排列,假设结果为 $R'=\{r'_1,r'_2,\cdots,r'_s\}$,其中 $s=|U'|$;

(3) 令 $U/\text{class}=\{g^{-1}(g(x))\mid x\in U\}$; //$U/\text{class}$ 表示语义类的集合

(4) for each $X\in U/\text{class}$ //对各类进行处理

(5) { 令 $tmp=\varnothing$;

(6) for each $x\in X$ do

(7) { 令 $t=\Re(x)$;

(8) if 存在 $t'\in tmp$,使得 $t\subseteq t'$, then 令 $tmp=tmp-\{t'\}\cup t$;

(9) else if 存在 $t'\in tmp$,使得 $t'\subset t$, then $tmp=tmp$; // tmp 不变

(10) else $tmp=tmp\cup t$; // $t'\not\subseteq t$ 且 $t\not\subseteq t'$

}

(11) for $i=1$ to $|U'|$ do

(12) { for each $t\in tmp$ do if $|t|>1$ then $t=t-\{r_i\}$;

(13) for each $t\in tmp$ do if 存在 $t'\in tmp-\{t\}$,使得 $t\subseteq t'$, then 将 t' 从 tmp 中删除; //吸收律

}

(14) for each $t\in tmp$ do $U''=U''\cup t$;

(15) }.

实测试表明,算法中 tmp 的规模几乎是常数(几乎不随 $|U|$ 变化),所以整个算法的复杂度约为 $O(|U/\text{class}\|X\|U|n')\approx O(n'|U|^2)$。

3.3 相容参数的自适应优化

每一特征 a_i 都有对应的相容参数 λ_i。令 $\min_i$ 和 $\max_i$ 分别表示特征 a_i 的最小和最大特征值。当 λ_i 被设置为 $\max_i-\min_i$ 时,所有对象被视为同属于一个关于 a_i 的相容粒,这样所有对象关于 a_i 是不可区分的,从而使得 a_i 失去分辨能力而造成"人为落选"。当 λ_i 被设置为 0 时,相当于利用特征 a_i 的原始分辨能力,这样会导致极少数的特征被选择,甚至仅有特征 a_i 被选择,最终造成过学习而降低分类器的泛化能力。因此,关键是如何在 $(0,\max_i-\min_i)$ 中选择适当的相容参数。

令 $\lambda_i=\mu(\max_i-\min_i)$,其中 $\mu\in(0,1)$,μ 称为相容度。不难发现,存在 $\mu_C\in(0,1)$,使得当 $\mu\leqslant\mu_C$ 时,所得到的相容粒度空间是一致的,当 $\mu>\mu_C$ 时则是不一致的。这样的 μ_C 称为一致临界点。此外,如果 μ 取值为 μ_M,分类器获得最大的分类准确率,则将 μ_M 称为最大值点。

容易推知,当 μ 从 0 向 1 变化时,相容粒由最初的单个对象构成的集合逐步变大,以至于最后所有的相容粒都变到最大——都等于 U。类似于定理 3 的分析可推知,相容粒的这种变化过程是非递减过程,因此,一致临界点 μ_C 关于一致性是唯一的。显然,μ_C 和 μ_M 不会出现在两个端点($\mu=0$ 和 $\mu=1$)

上,在其附近一般也极少出现,同时还可能存在多个最大值点。这给我们寻找最大值点带来了困难。

幸运的是,在多次的实际测试中我们发现,第一个最大值点一般出现在一致临界点的右附近,而且是右靠近一致临界点的第一个极大值点。据此观察,可通过比较 $|\rho_{T(F)}U)|$ 是否等于 $|U|$ 的方法先找出一致临界点,然后从一致临界点起通过梯度爬行(从左到右)可以找出第一个极大值点,从而获得近似的最大值点。这样,就可以把众多的相容参数优化问题转化为一致临界点的查找和梯度爬行搜索问题,从而有效简化了问题的复杂性。

基于上述考虑以及利用前面提出的算法,下面给出完整的自适应图像语义分类算法。

算法 8 完整的分类算法.

输入:U, U_T, $F = \{a_1, a_2, \cdots, a_n\}$. //$U_T$ 用于相容参数的自适应优化

输出:规则集 R(分类器).

(1) 令 $gra\_val = val$; //设置梯度爬行步长,val 为(0,1)之间实数,如 0.01 等

(2) 扫描 U, 计算 $\max_i$ 和 $\min_i$, $i = 1,2,\cdots,n$;

(3) 令 $\mu = 0.1$, $\lambda_i = \mu(\max_i - \min_i)$, $i = 1,2,\cdots,n$;

(4) 令 $v = |\rho_{T(F)}(U)|$;

(5) while $v = |U|$ do //逼近一致临界点

(6) {$\mu = \mu + gra\_val$;

(7) $\lambda_i = \mu(\max_i - \min_i)$, $i = 1,2,\cdots,n$;

(8) 令 $v = |\rho_{T(F)}(U)|$; //改变值的 λ_i 将影响着 v 的值

}

(9) 令 $\mu = \mu - gra\_val$; //确定一致临界点,此后根据需要也可重新设置更小的 gra_val

(10) $acc_1 = 0$, $acc_2 = 0.0001$, $U'' = U$, $\lambda_i = \mu(\max_i - \min_i)$, $i = 1,2,\cdots,n'$;

(11) while $acc_1 < acc_2$ do //寻找右靠近一致临界点的第一个极大值点

(12) {令 $R = U''$;

(13) 进行特征选择,设结果为 $F' = \{a_1, a_2, \cdots, a_{n'}\}$; //利用算法 5

(14) 基于 F' 构造规则集 U'; //利用算法 6

(15) 对 U' 进行简化,得到分类器 U''; //利用算法 7

(16) 令 $acc_1 = acc_2$;

(17) 使用 U_T 对分类器 U'' 进行准确率测试,并把结果赋给变量 acc_2;

(18) $\mu = \mu + gra\_val$;

(19) $\lambda_i = \mu(\max_i - \min_i)$, $i = 1,2,\cdots,n'$;

}

(20) 返回 R.

在该算法中,步骤(13)的计算时间为 $O(n(n-1)|U|\log(|U|))$。步骤(14)至(17)的计算时间约为 $O(n'^2|U|^2)$。但在实验中发现 n' 一般远小于 $n = |F|$,当 $|U|$ 不是很大时,步骤(14)至(17)相对于(13)来说,几乎是“瞬时”完成的。也就是说,步骤(13)是最耗时的,它几乎决定了整个算法的执行时间。因此,步骤(12)至(19)的计算时间约为 $O(n(n-1)|U|\log(|U|)) \approx O(n^2|U|\log(|U|))$。整个算法的复杂度可以表示为 $O((1/gra\_val)n^2|U|\log(|U|))$。该复杂度具有较大的弹性,主要跟 gra_val 的设置有关。如果 gra_val 设置得很小,那么相容参数的优化效果就会很好,但这样会增加整个算法的计算时间;如果 gra_val 设置得很大,那么算法的计算时间会减少,但是所得分类器的分类效果和泛化能力将会相对差些。

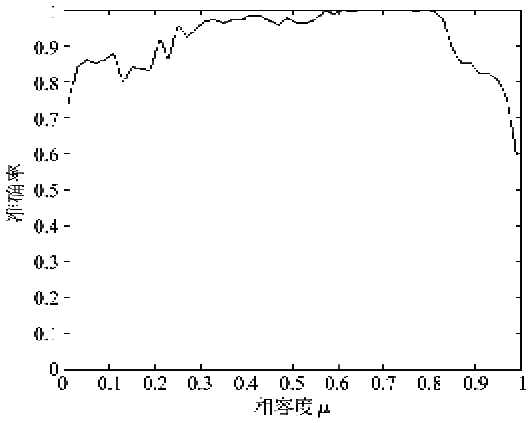
需要指出的是,算法 8 采用了一个在实验中统计出来的假设:右靠近一致临界点的第一个极大值点为最理想的最大值点。虽然在实验中还没有发现存在推翻这个假设的证据,但目前我们还没有在理论上证明这个假设是正确的,因此为保险起见,可在算法 8 的步骤(11)上增加一些循环条件,如 $\mu \leqslant 0.6$ 等,以保证 μ 不取值到 1 的附近。

4 实验分析

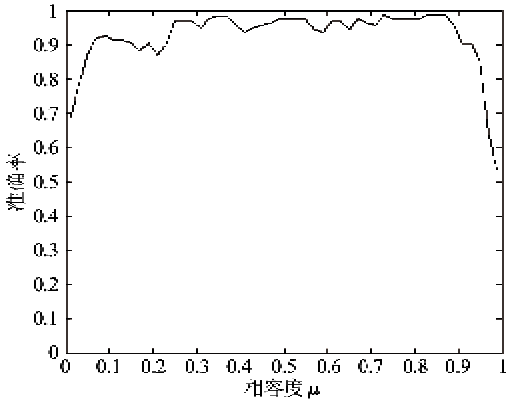
为验证提出算法的性能,从 Corel 图像库和 http://wang.ist.psu.edu/docs/related/上下载了 4 种语义类图像,分别是 tree, building, flower, mountain,其中 tree 包含 136 幅图像,其他均包含 100 幅图像。主要根据文献[19],对每幅图像提取 22 个特征。

首先考察分类准确率随相容度 μ 变化的情况。我们将 mountain 和 tree、building 和 flower 以及 elephant 和 bird 分别组成 3 组训练集。在提取特征时,对每一图像类,按照 1:1 分别构造训练集和测试集(本文中,训练集和测试集都不相交)。然后观察分类准确率与相容度之间的变化关系,结果如图 1 所示,图中纵向虚线对应的横坐标值为测出来的近似一致临界点。图 1 表明通过不断调节相容度的方法都可以在这些数据集上获得很高的分类准确率(图 1(a)、(b)和(c)的最高准确率分别为 100%、98.25%和 100%);同时,最大值点(可能存在多个)通常分布在一致临界点的右边,而且第一个最大值点(从左往右数)就是右靠近一致临界点的第一个

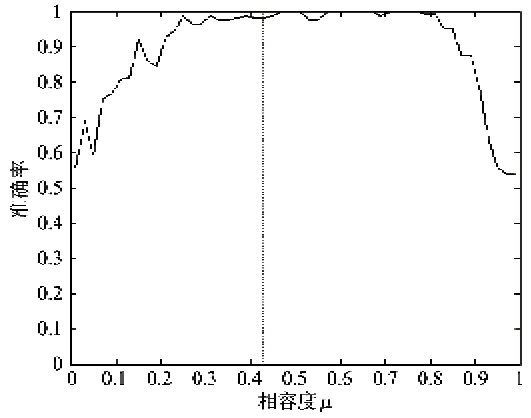
极大值点(至少在我们多次的实验中都证实了这一点)。由于判断一个点是否为极大值点比直接计算在这个点上的分类准确率要省时得多,所以在搜索最大值点时,先找出一致临界点,然后从一致临界点起向右梯度爬行,找出第一个极大值点,并以此作为近似的最大值点。



(a) mountain 和 tree 训练集



(b) building 和 flower 训练集



(c) elephant 和 bird 训练集

图 1　分类准确率随相容度 μ 的变化情况

此外,在一致临界点的右边可能存在多个最大值点。但离一致临界点越远的最大值点,其导致的相容粒度空间的不一致程度就越大。这可以从图 2 中得到证实,其中训练集由 mountain 和 tree 构成。语义类对特征集 F 的依赖程度 $\gamma_{T(F)}(U)$ 的减小意味着正区域 $\rho_{T(F)}(U)$ 在变小,这说明相容粒度空间的不一致程度在加大。当 $\gamma_{T(F)}(U)$ 减小到一定程度时,分类器会严重偏离训练集所蕴涵的客观规律。这时虽然分类器仍然能够覆盖测试集,但这种覆盖因"过泛"而缺乏针对性,从而可能导致失去应有的应用价值。如果强制 $\gamma_{T(F)}(U)$ 等于 1,那么分类器的预测能力将限于训练集所界定的范围,从而导致其泛化能力不强。因此,允许 $\gamma_{T(F)}(U)$ 略小于 1 是合理和可行的。这说明,右靠近一致临界点的第一个极大值点为最理想的最大值点。因此,从一致临界点起,向右进行梯度搜索,以找到最靠近一致临界点的第一个极大值点,并以此作为近似的最大值点,这种做法是可行的。

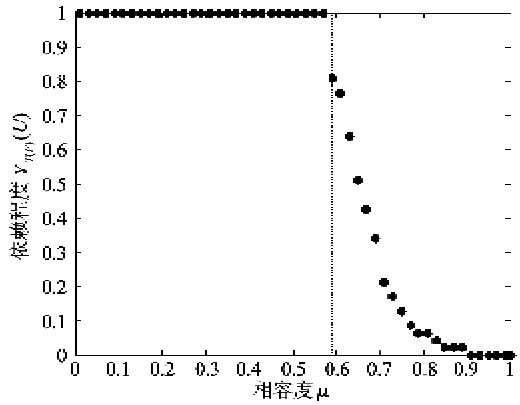


图 2　依赖程度 $\gamma_{T(F)}(U)$ 随相容度 μ 变化的情况

算法 8 中爬行步长 *gra_val* 和相容度 μ 的取值对分类结果有较大的影响。表 1 展现了分类准确率随 *gra_val* 和 μ 的变化情况,其中,样本集是由 tree 和 flower 构成,训练集和测试集规模之比也是 1:1。从表 1 可以看出,算法 8 首先使用步长 *gra_val* 为 0.02来寻找一致临界点,结果发现 $\mu = 0.53$ 为一致临界点。此后,为了增加求解精度,使用更小的爬行步长0.005进行搜索。当 $\mu = 0.570$ 时,分类准确率为0.9841,这比前面的结果 1.0($\mu = 0.565$)小。这表明,在此步长下 $\mu = 0.565$ 是右靠近一致临界点的第一个极大值点,因此其对应的值即为最佳的分类准确率。

从表 1 也可进一步看出,参数 *gra_val* 对分类结果的控制是通过修改 μ 来实现的。这分为两个过程。第一个过程是为了求一致临界点,对应算法 8

中的步骤(1)至(9)。由于此过程中每次循环只需计算 $|\rho_{T(F)}(U)|$，循环次数不超过 $1/gra\_val$ 次，速度相对比较快，因此可以设置得小一些；第二个过程是在第一个过程的基础上对最大值点进行逼近，对应算法 8 中的步骤(10)至(19)。这时 gra_val 设置得越小，就越能逼近最大值点(分类精度也就越高)，但计算时间就越长。因此 gra_val 的设置需要在计算时间和分类精度之间进行折中，且与数据集的规模有关。针对本文涉及的几个数据集来说，我们认为 gra_val 取0.01比较适合。

表 1 分类准确率随 gra_val 和 μ 的变化情况

μ	$\|\rho_{T(F)}(U)\| = \|U\|$?	μ	分类准确率
0.01	是	0.530	0.9159
0.03	是	0.535	0.9206
0.05	是	0.540	0.9206
0.07	是	0.545	0.9788
0.09	是	0.550	0.9841
0.11	是	0.555	0.9841
…	…	0.560	0.9894
0.53	是	0.565	1.0000
0.55	否(不一致)	0.570	0.9841
$gra\_val = 0.02$		$gra\_val = 0.005$	

为验证提出算法的分类性能，进一步从 Corel 图像库下载了 16 类图像，其语义类标识分别为 antelope, barbecue, building, dog, firearm, firework, food, homeitem, men, NewYork, plantart, plant, polo, sailing, women, workship，每个语义类都是 100 幅图像。

考虑到多类别分类问题，我们对每幅图像提取 51 个特征[19]，并按照 1:9 构造训练集和测试集。这样，训练集和测试集的规模分别为 160 和 1440。我们分别用最近邻分类、高斯贝叶斯分类、SVM 分类等与本文的方法进行比较，结果如表 2 所示。

表 2 比较结果(训练集:测试集 =1:9，不相交)

分类方法	最近邻分类	高斯贝叶斯分类	SVM 分类	SVM 分类	本文方法(算法 8)
分类准确率	31.88%	24.31%	31.60%	25.83%	34.79%
说明	使用原始特征	用 PCA 投影，主向量取 9 维	使用原始特征，维缩放至 [-1,1]	使用原始特征，并未经缩放	使用原始特征

表 2 说明，在原始特征数据上，本文提出的方法均比其他同类方法获得更好的分类准确率。但注意到，所有方法获得分类准确率的绝对数值都比较低。我们认为，除主要因为类别增多(一共 16 类)以及训练集和测试集规模之比偏低(1:9)等因素以外，这一结果还跟所使用的样本集有关。例如，从图像集中我们看到，building 类(建筑物)和 NewYork 类(纽约)中的图像是较难区分的，它们呈现出来的大多都是建筑物；此外，plantart(植物艺术品)和 plant(植物)、sailing(航海)和 workship(船舶)、women(女人)和 men(男人)等也有类似的问题。这进一步说明，即使是在干扰性很强的图像集中本文方法亦获得了相对较好的分类效果。

5 结 论

本文将相容粒度空间模型引入图像语义分类中，提出了一种自适应的图像语义分类方法，由此构造的自适应分类器可以在一定程度上实现底层视觉特征与高层语义空间之间的非线性映射，为解决语义鸿沟问题探索了一种途径。该方法可以直接对连续型数值特征进行处理，而不需要对其进行离散化等预处理操作，从而在一定程度上避免因预处理而造成的有用信息的丢失和图像信息结构的人为改变。通过相容参数的自适应优化，本文提出的方法几乎不需要手工进行参数设置即可自动寻找最佳的相容粒度空间并在此空间中形成对语义类的最佳粒度表示，从而自动适应于不同应用背景下的图像语义分类。针对原始特征为数值型特征并且特征值在局部上相对“连续”的特征空间，这种方法具有独特的优势，一般比同类方法具有更好分类效果，因而具有一定的推广和应用价值。

今后，将进一步探讨最大值点、极大值点和一致临界点之间的关系(这种关系是否可量化，如何量化？等)；此外，在相容参数自适应优化方面仍然存在许多可改进的地方，相容参数之间也可能存在一些可量化的依赖关系。在此基础上，将此方法应用于图像语义标注，并探讨复杂图像环境下的语义检索问题以及现实对图像的深层次语义理解。

参考文献

[1] Smeulders A W M, Worring M, Santini S, et al. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000, 22 (12): 1349-1380

[2] 李志欣, 施智平, 李志清等. 融合语义主题的图像自动标注. 软件学报, 2011, 22 (4): 801-812
[3] 李志欣, 施智平, 李志清等. 图像检索中语义映射方法综述. 计算机辅助设计与图形学学报, 2008, 20 (8): 1085-1096
[4] Vailaya A, Yu Z, Jain A K. A hierarchical system for efficient image retrieval. In: Proceedings of the 13th International Conference on Pattern Recognition, Washington, USA, 1996. 356-360
[5] Vailaya A, Figueiredo M, Jain A, et al. A Bayesian framework for semantic classification of outdoor vacation images. In: Proceedings of SPIE: Storage Retrieval Image Video Databases VII, vol. 3656, San Jose, USA, 1999. 415-426
[6] 王崇骏, 杨育彬, 陈世福. 基于高层语义的图像检索算法, 软件学报, 2004, 15 (10): 1461-1469
[7] Park S B, Lee J W, Kim S K. Content-based image classification using a neural network. *Pattern Recognition Letters*, 2004, 25 (3): 287-300
[8] Kang S, Park S. A fusion neural network classifier for image classification. *Pattern Recognition Letters*, 2009, 30: 789-793
[9] Chapelle O, Hafther P, Vapnik V. Support vector machines for histogram based image classification. *IEEE Transactions on Neural Networks*, 1999, 10 (5): 1055-1064
[10] 万华林, Morshed U C. Image semantic classification by using SVM. 软件学报, 2003, 14 (11): 1891-1899
[11] 黄启宏, 刘钊. 基于多平面支持向量机的图像语义分类算法. 光电工程, 2007, 34 (8): 99-104
[12] 廖绮绮,李翠华. 基于支持向量机语义分类的两种图像检索方法. 厦门大学学报(自然科学版), 2010, 49 (4): 487-494
[13] Jiang Y, Chen J, Wang R S. Fusing local and global information for scene classification. *Optical Engineering*, 2010, 49 (4): 1-10
[14] 张向荣, 谭山, 焦李成. 基于商空间粒度计算的 SAR 图像分类. 计算机学报, 2007, 30 (3): 484-490
[15] Zheng Z, Hu H, Shi Z Z. Rough set based image texture recognition algorithm. *Lecture Notes in Computer Science*, 2004, 3213: 772-780
[16] Zheng Z, Hu H, Shi Z Z. Tolerance relation based information granular space. *Lecture Notes in Computer Science*, 2005, 3641: 682-691
[17] Shi Z Z, Meng Z Q, Yuan L. Tolerance granular computing based on incomplete information system. In: Proceedings of the 2009 IEEE International Conference on Granular Computing, Nanchang, China, 2009. 501-506
[18] Meng Z Q, Shi Z Z. A fast approach to attribute reduction in incomplete decision systems with tolerance relation-based rough sets. *Information Sciences*, 2009, 179: 2774-2793
[19] 施智平, 胡宏, 李清勇等. 基于纹理谱描述子的图像检索. 软件学报, 2005, 16 (6): 1039-1045

Self-adaptive image semantic classification based on tolerance granular space model

Meng Zuqiang*, Shi Zhongzhi**
(*College of Computer, Electronics and Information, Guangxi University, Nanning 530004)
(**Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 1001900)

Abstract

Aiming at the problem of the semantic gap between the low-level feature and the high-level semantic, the paper uses the tolerance granular space model to study image semantic classification, and then proposes a self-adaptive image semantic classification method, thus an effective way for solving the semantic gap problem is given. The proposed method models an image set as a primitive feature-based tolerance granular space, in which the tolerance relation is defined by using tolerance parameters and establishing a distance function, and then the size of an object's neighborhood granule can be controlled effectively and finally the real-valued features can be directly dealt with without any pretreatment, such as discretization. In addition, tolerance parameters can be self-adaptively optimized by introducing the concept of tolerance degree, so as to automatically control the size of an object's neighborhood granule, and in this way, the obtained classifier can adjust itself to a variety of image sets almost without any manual parameter configuration. The experimental results show that the proposed method is effective and feasible, and it has better classification performance than that of similar methods.

Key words: image semantic classification, granular computing (GrC), self-adaptation, tolerance granular space, tolerance relation