

“图像感知、手写识别” 读书报告

姓名：金泽文 学号：PB15111604

摘要：

在物联网走进人们生活的同时，产生了新的搜索需求与搜索技术。“以图搜图”，“手写识别”等技术开始出现并发展，成为物联网领域中不可忽视的两块领域。本报告通过对以图像感知哈希算法为例，对“以图搜图”技术展开分析，并通过基于BP神经网络的手写数字识别技术为例，对“手写识别”技术进行阐述。

关键词：

图像感知、感知哈希算法、手写识别、BP神经网络

1. 前言

如今，物联网的应用非常广泛，已经渗入到人们的学习、工作和生活中。随着这些物联网系统的开发和应用，人们越来越需要准确、智能地搜索现实世界中的物理实体信息，越来越需要不同于只针对静态计算机文字编码的传统搜索的更加强大的新的搜索功能。同时在物联网时代，处处搜索、时时搜索、物物搜索的发展方向上，手写识别成为拍摄扫描搜索的必备要素。本报告针对近来的谷歌等的“以图搜图”技术，与有道词典等的“手写识别”并搜索的查词技术展开阐述与分析。

2. 图像感知与手写识别

2.1 图像感知哈希算法

2.1.1 概述

图像感知哈希（Perceptual Hashing）是一门新兴技术，它通过对图

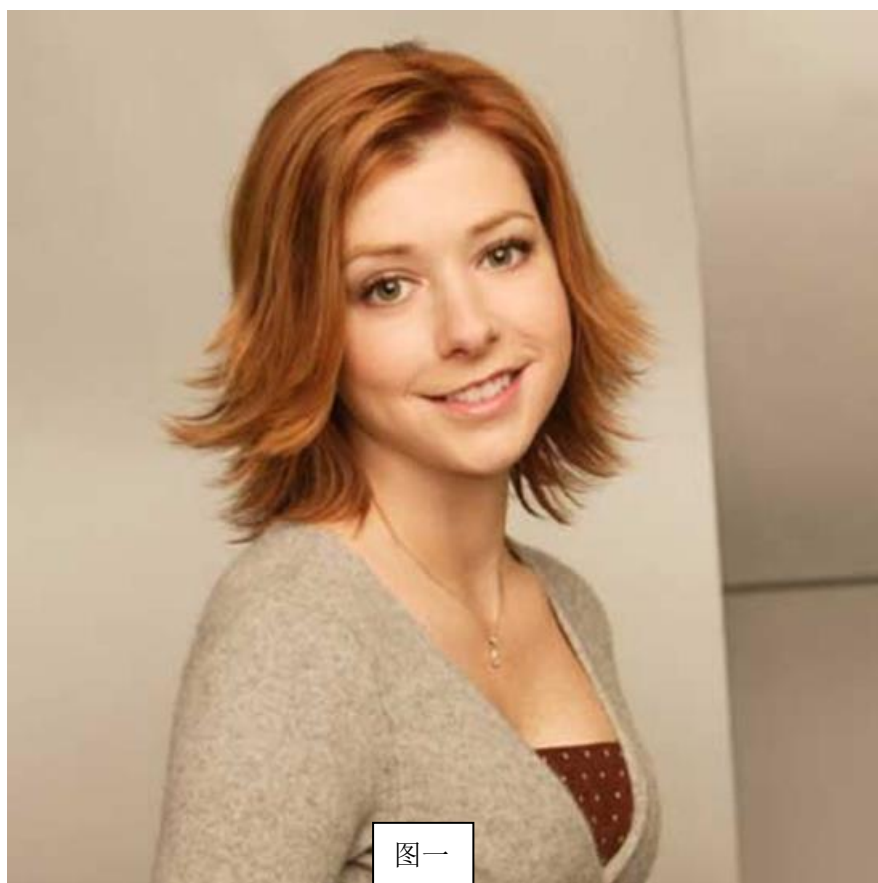
像感知信息的简短摘要和基于摘要的匹配，来支持图像的认证和识别，具有广泛的应用前景[1]。感知哈希通过对媒体感知内容和信息的提取，压缩形成一个简短的摘要，用于记录或者标识媒体内容，或者更进一步用于发现媒体内容遭受的篡改或篡改的位置。图像感知哈希是感知哈希的一大分支，其特征提取和编码阶段需要包含更多的视觉感知信息，以满足其鲁棒性要求。

2.1.2 算法

以一个较为简单的利用灰度的实现为例。

第一步，缩小尺寸。

将图片缩小到 8×8 的尺寸，总共 64 个像素。这一步的作用是去除图片的细节，只保留结构、明暗等基本信息，摒弃不同尺寸、比例带来的



图一



图二



图三

图片差异。如图一（原图），图二（缩小后），图三（便于观察缩小后）

第二步，简化色彩。

将缩小后的图片，转为 64 级灰度。也就是说，所有像素点总共只有 64 种颜色。如图四，图五：



图四



图五

第三步，计算平均值。

计算 64 个像素的灰度平均值。

第四步，比较像素的灰度。

将每个像素的灰度，与平均值进行比较。大于或等于平均值，记为 1；小于平均值，记为 0。

第五步，得到哈希值。

将上一步的比较结果，即 0、1 串组合在一起，就构成了一个 64 位的整数，即为哈希值。

第六步，比较哈希值。

根据两个图片哈希值的相似度进行比较，哈希值越相似，则原图越可能相似。

2.1.3 分析

这一算法的实现极其简单，很容易编程实现，并且比较的过程快速简单，同时不受图片大小缩放的影像。缺点也很明显，图片不能有某些变更，可能一些轻微的变化，就会导致截然不同的结果。所以，实际应用

过程中为了更好的结果与更好的鲁棒性，一般会采用更强大的算法。比如，Fridrich 利用图像 DCT 低频系数的不变性来获得图像信息的稳定编码[2]；而 Vishal Monga 等人提出的一组图像感知哈希算法分别利用了图像的特征点(feature points)不变性[3]、图像线性操作的矩阵不变性(matrix invariants)[4]以及图像小波分解得到的低通部分的灰度平均值[5]等。

2.2 基于 BP 神经网络的手写识别技术

2.2.1 概述

近来，神经网络在各个领域开始大量应用，并且在自动控制、模式识别、数据挖掘等方面大显身手。

BP(back propagation)神经网络是 1986 年由 Rumelhart 和 McClelland 为首的科学家提出的概念[6]，是一种按照误差逆向传播算法训练的多层前馈神经网络，是目前应用最广泛的神经网络。

2.2.2 算法：

网络层数：解决非线性数据的问题，单层网络的精度不够高，增加网络层数能提高精度，降低误差，但是也带来了网络复杂、学习缓慢等问题。基于上述理由，选择含有一个隐藏层的双层 BP 神经网络结构，可以较好地达到预期目标。

数据收集：考虑到程序学习速度、计算机性能和效率问题，输入图片统一使用 20*20 像素的白底图片来模拟输入框范围，数字在白底上写上 0 到 9 的数字以模拟移动设备上的手写输入。

输入输出：对于输入层，由于对收集的数据采用黑白两色，所以针

对 20*20 像素的处理，输入层需要 400 个神经元。对于输出层，由于只需要 0-9 有 10 个数字，所以设置 10 个神经元。

隐含层神经元个数：由理论研究可知，具有单隐层的三层神经网络足够执行任意复杂的函数映射。而实验表明，如果隐层结点数过少，网络不能具有必要的学习能力和信息处理能力。反之，若过多，不仅会大大增加网络结构的复杂性，网络在学习过程中更容易陷入局部极小点，而且会使网络的学习速度变得很慢。在权衡上述因素后，本文选择了经验公式：

$$h = (i * (o + 1) + 1) ^ 0.5.$$

其中，h 代表隐含层节点数，i 代表输入层节点数，o 代表输出个数，由此确定隐含层节点为 66 个。

激活函数的选择：采用 Sigmoid 函数：

$$y = 1 / (1 + e ^ (-x/p)).$$

这里 p 取 1。

学习速率：在 BP 神经网络中，学习速率对整个系统的影响十分大，学习速率过快会导致权值变化幅度大，权值修正时可能越过收敛点而导致系统震荡，无法得到期望权值；学习速率过慢又会导致收敛速度过慢，浪费计算机的资源和时间。因此选择一个合适的学习速率非常重要。学习速率的范围一般在 0.01 到 0.8 之间，这里将学习速率确定为 0.1，较小的学习速率收敛速度虽然会比较慢，但是能够保证系统稳定。

初始权值：由于本文的激活函数选取的是 Sigmoid 函数，当神经

元输入加权后的值接近于 0 的时候，权重的调节速率最大，因此要保证神经元输入权值较小，使每个权值在-0.1 到+0.1 之间随机生成，保证每个神经元的权值都能够在 Sigmoid 函数变化最大之处进行调节。

期望误差值：期望误差值越小需要的学习时间越长；期望误差值过大则系统不会达到稳定，识别效果不好。期望误差值设定为 10^{-5} 比较合适。

训练过程：采集样本图片随机生成 400×66 个隐藏层权重→保存权重值→得到 66 个隐藏层输出→随机生成 66×10 个输出层权重→保存权重值→得到 10 个输出层输出→计算误差→计算残差→残差乘以输出层权重→计算隐藏层残差→结果乘以输入值和学习速率→得到权值变化量→改变权值→进行下一轮修正。

当输出层的输出与样本值的误差在 10^{-5} 以内时，训练完成；考虑到网络不一定收敛，设定 1000 次最大循环次数，训练循环超过 1000 次则视为失败，此时应当改变学习速率、初始权值等各参数，再次进行学习。

2.2.3 分析：

基于 BP 神经网络的手写识别技术整体效果良好，但是由于输入样本中的个别坏样本，以及选择不恰当的层数，还有激活函数的过于简单，毫无针对性等等原因会有较小的出入。正常的识别率可以达到 90%以上。然而，BP 算法也有一定的缺陷[7]：对初始权重的极度敏感，很容易导致结果收敛于局部极小区域；经常滞留在误差梯度曲面的平坦区，导致收敛缓慢甚至无法收敛。由于这些问题，单纯的 BP 神经

网络也许并不适合手写识别，只有改用其他网络，或者对 BP 网络进行大幅度的改进，才可能带来质的飞跃。

3. 小结

物联网时代中的搜索将是与传统搜索截然不同的概念，全新的功能拓展、维度变化将是接下来几年发展的一个重要要素。而以现在“以图搜图”、“手写识别”正在作为新时代搜索技术的代表飞速发展。这次通过对图像感知算法，以及 BP 网络下的手写识别的调研，进一步理解了计算机科学，或者叫人工智能领域的进步与发展。物联网时代才刚刚到来，人工智能时代也才刚刚开始，一切的智能都还不够成熟，万物互联的美景才显露一丝风貌。

参考文献

- [1] Hadmi A, Puech W, Said B A E, et al. Perceptual image hashing[M]. Watermarking-Volume 2. InTech, 2012.
- [2] Fridrich J, Goljan M. Robust hash functions for digital watermarking[C]. Information Technology: Coding and Computing, 2000. Proceedings. International Conference on. IEEE, 2000: 178-183.
- [3] Monga V, Evans B L. Robust perceptual image hashing using feature points[C]. Image Processing, 2004. ICIP'04. 2004 International Conference on. IEEE, 2004, 1: 677-680.
- [4] Kozat S S, Venkatesan R, Mihçak M K. Robust perceptual image hashing via matrix invariants[C]. Image Processing, 2004. ICIP'04.

2004 International Conference on. IEEE, 2004, 5: 3443-3446.

[5] Venkatesan R, Koon S M, Jakubowski M H, et al. Robust image hashing[C]. Image Processing, 2000. Proceedings. 2000 International Conference on. IEEE, 2000, 3: 664-666.

[6] Plaut D C. Experiments on Learning by Back Propagation[J]. 1986.

[7]张铃, 张钊. 神经网络中 BP 算法的分析[J]. 模式识别与人工智能, 1994, 7(3): 191-195.