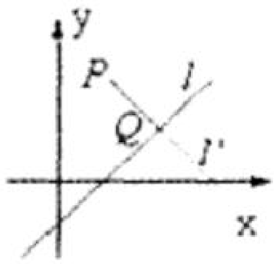
1. 证明：点 (u,v) 到一条线 (a,b,c) 的距离为：$|au+bv+c|$，这里 $a^2+b^2=1$ （15 分）

点 P 到直线 l 的距离是点 P 到直线 l 的垂线段的长，设点 P 到直线 l 的垂线为 l’ 垂足为 Q，由 l 垂直于 l’ 可知 l’ 的斜率为 B/A。



所以，l’的方程为：$y-y_0=\frac{B}{A}(x-x_0)$ 与 l 连立方程组，可以解得交点 $Q(\frac{B^2x_0-ABy_0-AC}{A^2+B^2},\frac{A^2y_0-ABx_0-BC}{A^2+B^2})$，则有：

$$
\begin{aligned}
|PQ|^2 &= (\frac{B^2x_0-ABy_0-AC}{A^2+B^2}-x_0)^2+(\frac{A^2y_0-ABx_0-BC}{A^2+B^2}-y_0)^2 \\
&= (\frac{-A^2x_0-ABy_0-AC}{A^2+B^2})^2+(\frac{-B^2y_0-ABx_0-BC}{A^2+B^2})^2 \\
&= \frac{A^2(Ax_0+By_0+C)^2+B^2(Ax_0+By_0+C)^2}{(A^2+B^2)^2} \\
&= \frac{(Ax_0+By_0+C)^2}{(A^2+B^2)}
\end{aligned}
$$

所以：$|PQ|=\frac{|Ax_0+By_0+C|}{\sqrt{A^2+B^2}}$

代入：(u,v) 和 $a^2+b^2=1$，则有 $d=|Au+Bv+C|$

2. 简述 EM 算法的基本原理和流程（以高斯混合模型求解为例）（15 分）

由于我并没有修完先修的概率统计课程，我反复看了相关的 EM 理论推导网课[4]，希望能够阐述清楚这个问题。

首先我想回答几个学习过程中遇到的问题: 为什么要使用 EM 算法求解混合高斯模型？

因为 MLE（极大似然估计）无法得到似然函数的解析解。所以我们需要用 EM 算法去迭代求解，并且 EM 算法非常适合含有隐变量的概率模型的参数求解。

首先，根据联合概率分布的性质，我们很容易得到 $P(X,Z)=P(X)P(Z\mid X)$，（联合概率分布等于边缘概率分布乘以条件概率分布）。

移项并取对数有：

$$\log P(X)=\log P(X,Z)-\log P(Z\mid X)=\log\frac{P(X,Z)}{q(Z)}-\log\frac{P(Z\mid X)}{q(Z)}$$

等式两边同时关于 $q(Z)$ 求期望。

左边等于 $\int_Z q(Z)\cdot\log P(X)dZ=\log P(X)\int_Z q(Z)dZ=\log P(X)$

右边等于：

$$\int_Z q(Z)\log\frac{P(X,Z)}{q(Z)}dZ-\int_Z q(Z)\log\frac{P(Z\mid X)}{q(Z)}dZ=ELBO+KL(q\|p)$$

根据相对熵（KL 距离）的概念可知：$KL(q\|p)\geq 0$，当且仅当 $q=p$ 时取等号。

故要得到参数 θ 的最优估计，转化为最大化 ELBO(Evidence Lower Bound，证据下界)的问题。即：

$$\hat{\theta}=\arg\max_\theta ELBO=\arg\max_\theta\int_Z q(Z)\log\frac{P(X,Z)}{q(Z)}dZ$$

$$=\arg\max_\theta\int_Z P(Z\mid X,\theta^{(t)})\log\frac{P(X,Z)}{P(Z\mid X,\theta^{(t)})}dZ$$

$$=\arg\max_\theta\int_Z P(Z\mid X,\theta^{(t)})[\log P(X,Z)-\log P(Z\mid X,\theta^{(t)})]dZ$$

$$=\arg\max_\theta\int_Z P(Z\mid X,\theta^{(t)})\log P(X,Z)dZ$$

$$=\arg\max_\theta E_{Z\mid X,\ \theta^{(t)}}[\log P(X,Z)]$$

$$=\arg\max_\theta Q(\theta,\theta^{(t)})$$

E-Step

$$
\begin{aligned}
Q(\theta,\theta^{(t)}) &= \int_Z \log P(X,Z)P(Z\mid X,\theta^{(t)})dZ \\
&= \sum_Z \log\prod_{i=1}^{N} P(x_i,z_i)\prod_{i=1}^{N} P(z_i\mid x_i,\theta^{(t)}) \\
&= \sum_{z_1,z_2,\cdots,z_N}\sum_{i=1}^{N} \log P(x_i,z_i)\prod_{i=1}^{N} P(z_i\mid x_i,\theta^{(t)}) \\
&= \sum_{z_1,z_2,\cdots,z_N}[\log P(x_1,z_1)+\log P(x_2,z_2)+\cdots+\log P(x_N,z_N)]\prod_{i=1}^{N} P(z_i\mid x_i,\theta^{(t)}) \\
&= \sum_{z_1}\log P(x_1,z_1)\cdot P(z_1\mid x_1,\theta^{(t)})+\cdots+\sum_{z_N}\log P(x_N,z_N)\cdot P(z_N\mid x_N,\theta^{(t)}) \\
&= \sum_{i=1}^{N}\sum_{z_i}\log P(x_i,z_i)\cdot P(z_i\mid x_i,\theta^{(t)})
\end{aligned}
$$

在混合高斯模型中，已知：

$$P(X)=\sum_{k=1}^{K} p_k\cdot N(X\mid \mu_k,\Sigma_k)$$

$$P(X,Z)=P(Z)\cdot P(X\mid Z)=p_z\cdot N(X\mid \mu_k,\Sigma_k)$$

$$P(Z\mid X)=\frac{P(X,Z)}{P(X)}=\frac{p_z\cdot N(X\mid \mu_k,\Sigma_k)}{\sum_{k=1}^{K} p_k\cdot N(X\mid \mu_k,\Sigma_k)}$$

代入 Q 中，则有 $Q(\theta,\theta^{(t)})=\sum_{i=1}^{N}\sum_{z_i}\log p_{z_i}\cdot N(x_i\mid \mu_{z_i},\Sigma_{z_i})\cdot\frac{p_{z_i}\cdot N(x_i\mid \mu_{z_i},\Sigma_{z_i})}{\sum_{k=1}^{K} p_k\cdot N(x_i\mid \mu_k,\Sigma_k)}$

M-Step

$\theta^{(t+1)}=\arg\max_{\theta} Q(\theta,\theta^{(t+1)})$，也就是求 $p^{(t+1)}=(p_1{}^{(t+1)},p_2{}^{(t+1)},\cdots,p_K{}^{(t+1)})$

$$
\begin{cases}
\max_{p}\sum_{k=1}^{K}\sum_{i=1}^{N}\log p_k\cdot P(z_i=c_k\mid x_i,\theta^{(t)}) \\
\qquad\qquad s.t. \sum_{k=1}^{K} p_k=1
\end{cases}
$$

利用拉格朗日乘子法更新 P：

$$L(p,\lambda)=\sum_{k=1}^{K}\sum_{i=1}^{N}\log p_k \cdot P(z_i=c_k \mid x_i,\theta^{(t)})+\lambda(\sum_{k=1}^{K}p_k-1)$$

求偏导数并求解，可得：

$$\frac{\partial L}{\partial p_k}=\sum_{i=1}^{N}\frac{1}{p_k}P(z_i=c_k \mid x_i,\theta^{(t)})+\lambda \overset{\Delta}{=} 0$$
$$\Rightarrow \sum_{i=1}^{N}\sum_{k=1}^{K}P(z_i=c_k \mid x_i,\theta^{(t)})+\sum_{k=1}^{K}p_k\lambda=0$$
$$\Rightarrow \lambda=-N$$

故有 $p_k^{(t+1)}=\frac{1}{N}\sum_{i=1}^{N}P(z_i=c_k \mid x_i,\theta^{(t)})$，更新完成。

3. 用伪代码写出 Mean-shift 的算法流程（以图像分割为例），并分析影响算法性能的主要因素。（15 分）

我们的目标是对图像进行分割，也就是对于每个像素点需要给定一个类别的标签，这是我们的目的。在不考虑任何特征子的情况下，如果我们仅仅使用颜色空间作为特征。那么我们可以直接得到(X,Y,R,G,B)的五维特征。由于 RGB 是非均匀颜色空间[1]，我们可以选择转化为(X,Y,L,U,V)特征来保证颜色的均匀性。在 Mean-shift 迭代的过程中，可以人为指定超参数 Bandwidth 来确定高维球的半径，但是这样通常失去了泛化能力。在我参考了 Scikit-Learn 的 Mean-shift 代码后，发现可以通过数据来自动估计 Bandwidth[2]，这样的一般情况下是优于人为指定的情况的。需要说明的是作者在代码中提到，这样估计 Bandwidth 的算法复杂度为 $O(N^2)$，在样本较大时，可以随机采样部分样本来降低复杂度。每次迭代时，选取一个未标记的样本点，利用以 bandwidth 为半径的高维球中的样本点，计算漂移向量 $\vec{S}$。通常可以使用均值密度函数作为漂移向量的计算方法。值得一提的是，针对高维特征空间的“维度灾难”问题[3]，可以引入核函数方法有效降低运算的复杂度。在判断漂移向量 $\vec{S}$ 是否收敛时，阈值的大小也会影响运算的次数。

综上所述，影响算法性能的元素有：特征子的选取（128 维的 SIFT 特征和 5 维的 RGB 特征的复杂度肯定不同），估计 Bandwidth 的算法复杂度 $O(N^2)$，bandwidth 的大小，漂移向量的算法（高维特征空间中是否使用了合适的核函数优化），判断漂移向量 $\vec{S}$ 收敛的阈值等。

Algorithm 1 Mean-Shift

Input: Γ，Φ，B，t_1

/* Γ :*Picture features(X,Y,L,U,V),*
Φ :*Kernel,*
B :*Bandwidth*
t_1 :*threshold1*/*

Output: Λ

/* Λ :*Picture Segmentation(X,Y,k),* $k \in \{N_1, N_2, \cdots, N_K\}$ */

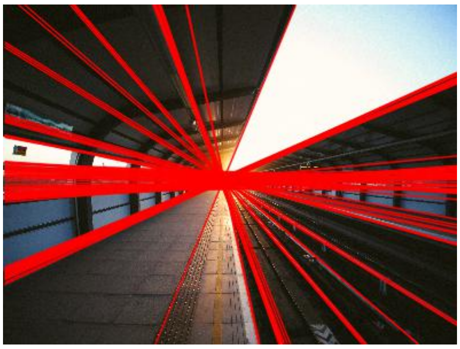
$i := 0$
do
  $C_i := (\Gamma / \Lambda)(random)$
  do
    for $\{P_i\}$ in sphere of C with bandwidth B **do**
      $\vec{S} := \sum f(P_j, C, \Phi)$
      $c(P_i, i) = c(P_i, i) + 1$
    end for
    $P = P + S$
  while $\|S\| > t_1$
  for $k \in \{1, 2, \cdots, j-1\}$ **do**
    if $\|C_k - C_i\| < \mathrm{B}$ **then**
      merge (C_k, C_i)
    end if
  end for
while not $\Gamma / \Lambda = \phi$
return Λ

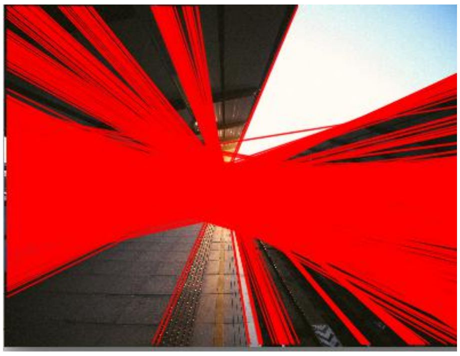
表一：Mean-Shift 算法的伪代码

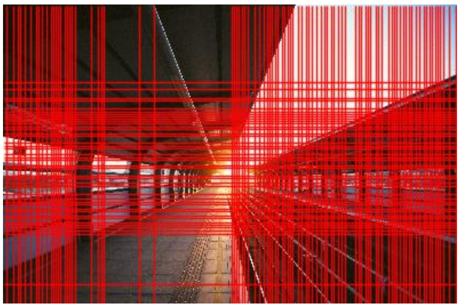
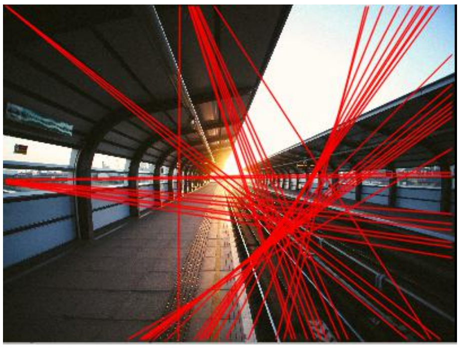
4. 找一张包含线条的图像，用霍夫变换进行线检测，并统计线条的数目。尝试不同的参数设置，并给出结果比较。（25分）

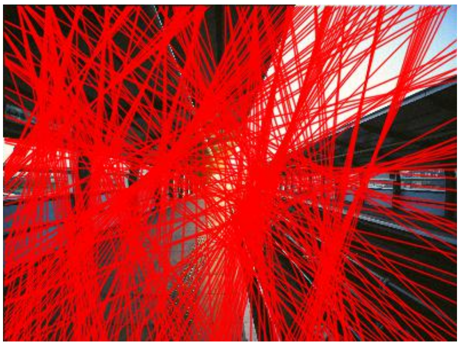
首先对原图进行灰度化，并使用 canny 作为边缘检测子。

	ρ	θ	min_points	lines

	1	1	215	1
	1	1	156	6
	1	1	145	15
	1	1	109	100

	1	1	64	754
	1	1	10	41286
	1	13	126	4
	1	13	73	48

	1	360	60	8
	1	360	2	133
	255	1	300	43
	156	1	300	39
	51	1	300	106

	26	1	300	292

可以看到 minpoints 越大，检测到的直线越少。θ 越大，检测直线时的角度步长越大，可能越来越多角度的直线检测不到。ρ 越大，检测走的像素步长越大，检测到的直线越少。

5. 用线拟合的方式，对下图中的各文字行，插入删除线

效果如下图所示，具体实现方法和技巧详见代码中的注释。

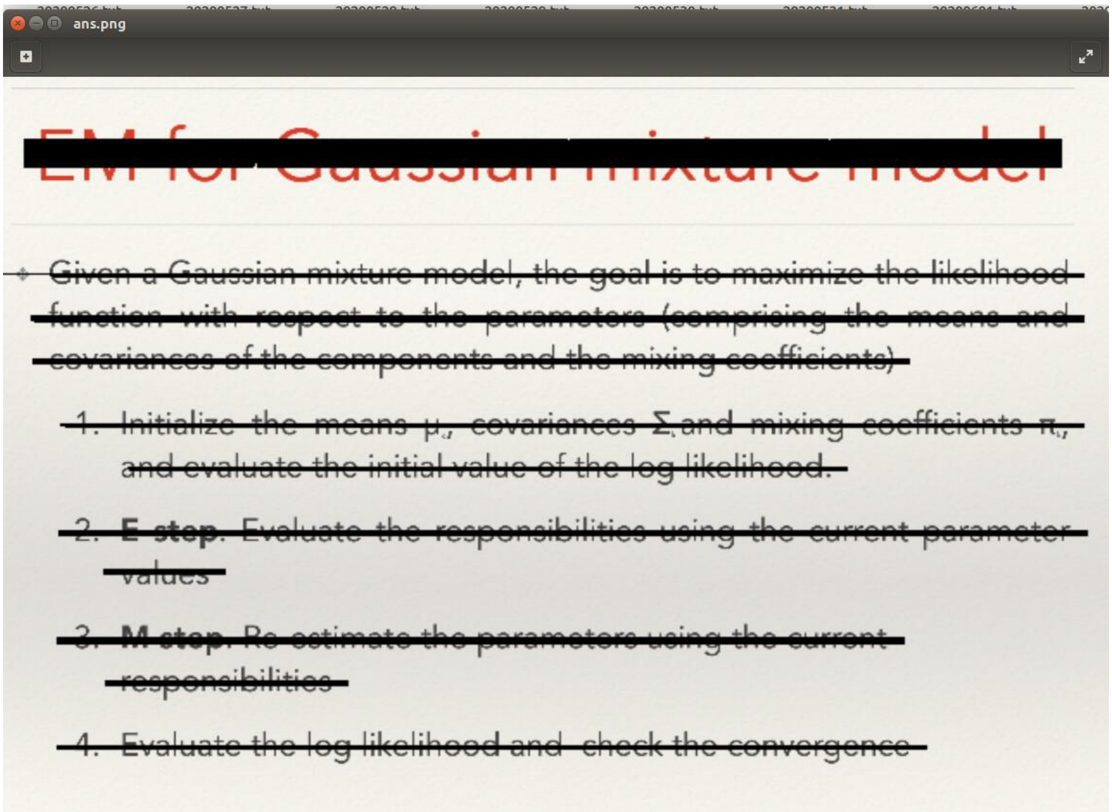


图 1 添加了删除线后的图片

参考文献

[1] 刘友明, 刘希顺, 刘安之,等. 一种基于 LUV 均匀颜色空间的彩色分割方法[J]. 微型电脑应用, 2000(12):27-28.

[2] 机器变得更残忍.scikit-learn 源码学习之 cluster.mean_shift.estimate_bandwidth[EB/OL].https://blog.csdn.net/jiaqiangbandongg/article/details/53495419,2016-12-06.

[3] 王华忠, 俞金寿. 核函数方法及其模型选择[C]// 第 17 届中国过程控制会议. 0.

[4] shuhuai008.【机器学习】【白板推导系列】【合集 1～23】[EB/OL].https://www.bilibili.com/video/BV1aE411o7qd,2019-10-11.