摘　要

由于多媒体数据的爆发式增长，本地设备的存储空间和计算能力不足以满足用户的需求，因此，图像外包成为了一种备受关注的解决策略。然而，由于图像数据包含大量的隐私信息，这些数据在传输、存储和共享过程中常常面临着泄露、篡改和恶意攻击等安全威胁。对于外包的图像，用户通常选择在外包前对图像进行加密以保证图像的安全性，并且希望在之后能对这些图像进行检索和使用。现有的图像检索方案大多都是基于内容的图像检索（Content-based image retrieval，CBIR），但明文域的 CBIR 方案难以应用于加密域。为了能够实现加密域上基于内容的图像检索，近年来提出了很多相关的方案，但加密域的 CBIR 方案常常会出现检索效率不高、检索准确度较低、存储代价太大等问题。因此，如何在保证图像安全的前提下实现高效、高准确度且存储代价小的检索方案成为了一个具有挑战性的问题。为了解决上述问题，本文针对图像可搜索加密算法进行了研究。本文的研究内容包含以下几个方面：

（1）提出了一种基于随机矩阵的特征加密算法，能够实现密文域的余弦相似度计算。为了保证图像特征的隐私性，对图像特征采用基于随机矩阵的特征加密算法，加密后的图像特征能够实现密文状态下余弦相似度的计算，并通过理论证明和实验验证了该算法的安全性以及密文下图像相似度计算的准确性。

（2）设计并实现了一个基于深度学习的图像可搜索加密原型系统。利用预训练的网络模型提取用于表征图像类别和图像特征的向量。利用类别标签构建了基于 K-means 聚类算法的层次索引树以提高检索效率。同时，采用了基于 LWE（Learning with Errors）的安全 kNN（k-Nearest Neighbors）算法保护索引的隐私安全。通过安全性分析和实验论证对系统性能进行了评估，实验结果表明本文提出的方案能够在多类别图像中实现较高的检索准确度和检索效率，且存储成本也是对比方案中最少的。

（3）提出了一种支持代理重加密的数据更新算法。在检索用户权限被撤销或者图像密钥被泄露时，图像所有者可以通过生成重加密密钥，委托云服务器实现图像数据的代理重加密，有效的减轻了图像所有者的计算负担。同时，本文还对已有类别增加图像和增加新类别图像两种情况下索引的更新进行了分析和实验。

关键词：深度学习；层次索引树；基于 LWE 的安全 kNN；余弦相似度；重加密

Abstract

Due to the explosive growth of multimedia data, the storage space and computing power of local devices are insufficient to meet the needs of users. Therefore, image outsourcing has become a widely recognized solution strategy. However, since image data contains a large amount of privacy information, these data often face security threats such as leakage, tampering, and malicious attacks during the transmission, storage, and sharing processes. For outsourced images, users usually choose to encrypt the images before outsourcing to ensure their security and hope to retrieve and use these images afterward. Existing image retrieval schemes are mostly based on content-based image retrieval (CBIR), but plaintext domain CBIR schemes are difficult to apply to the encrypted domain. In order to achieve content-based image retrieval in the encrypted domain, many relevant schemes have been proposed in recent years. However, encrypted domain CBIR schemes often suffer from problems such as low retrieval efficiency, low retrieval accuracy, and high storage cost. Therefore, how to achieve an efficient, accurate, and low-cost retrieval scheme while ensuring image security has become a challenging problem. To address the aforementioned issues, this thesis focuses on the research of image searchable encryption algorithms. The research content of this thesis includes the following aspects:

(1) A feature encryption algorithm based on random matrix is proposed, which enables cosine similarity calculation in the ciphertext domain. In order to ensure the privacy of image features, a feature encryption algorithm based on random matrix is applied to encrypt the image features. The encrypted image features can achieve cosine similarity calculation in the ciphertext state. The security of the algorithm and the accuracy of image similarity calculation in the ciphertext domain are theoretically proven and experimentally verified.

(2) A prototype system of image searchable encryption based on deep learning is designed and implemented. Pretrained network models are used to extract vectors for representing image categories and image features. A hierarchical index tree based on the K-means clustering algorithm is constructed using category labels to improve retrieval efficiency. In addition, a secure k-nearest neighbors (kNN) algorithm based on Learning with Errors (LWE) is employed to protect the privacy and security of the index. The system performance is evaluated through security analysis and experimental verification. The experimental results demonstrate that the proposed approach achieves high retrieval accuracy and efficiency in multi-category images, with the least storage cost compared to other schemes.

(3) A data update algorithm supporting proxy re-encryption is proposed. When a re-

trieving user's access rights are revoked or image keys are leaked, the image owner can generate a re-encryption key and delegate the cloud server to perform proxy re-encryption of the image data, effectively reducing the computational burden on the image owner. Additionally, this thesis analyzes and experiments with the updating of indexes in two scenarios: adding images to existing categories and adding images to new categories.

Key Words: Deep learning; Hierarchical index tree; Secure kNN algorithm based on LWE; Cosine similarity; Re-encryption

目 录

插图索引

附表索引

第 1 章　绪　论

1.1　研究背景和意义

随着智能手机、数码相机等成像设备的发展，图像呈现爆发式增长，而本地设备的存储空间和计算能力是有限的，因此更多用户选择将图像外包。然而图像一旦存储在云服务器中，用户就失去了对图像的直接控制权[1]，而图像中往往包含大量的隐私信息，如：医疗图像、军事图像、工业图像等，这些图像信息一旦泄露，将会对生命财产安全、国家安全、企业经济安全等方面造成影响[2]。2014 年，通信应用 Snapchat 的第三方图像处理服务提供商被黑客攻击，导致数十万用户的私人照片泄露[3]。2023 年，为英国政府机构提供服务的 IT 外包公司 Capita 遭受了一次黑客攻击，数据泄露造成的损失总额可达 1500 万至 2000 万英镑[4]。这些云安全事件的发生说明云服务存在着极大的安全隐患，因此对于图像信息的隐私保护刻不容缓。

为了保护图像中的隐私信息，通常采用图像加密再外包的方式，然而图像加密后会使得基于内容的图像检索变得困难。明文域的 CBIR 方案已经非常成熟，但这些方案无法直接应用于密文域。因此，如何在大规模密文图像数据集中快速、准确地检索到相似图像是一项极具挑战性的任务。尽管加密域图像检索已经得到广泛研究，但提出的各种方案仍存在准确度低、检索效率不高和存储开销较大等问题。

本文针对云计算中的图像可搜索加密算法展开研究。提出了基于随机矩阵的加密算法用于图像特征的隐私保护，该算法能实现密文域特征向量的余弦相似度计算和排序。本文设计并实现了一个基于深度学习的图像可搜索加密原型系统，该系统利用预训练的深度学习模型提取了用于表征类别标签和图像特征的向量，使用类别标签构建了基于 K-means 聚类的层次索引树以提高检索效率，并采用基于 LWE 的安全 kNN 算法对索引树加密，通过安全性分析和实验论证对方案的可行性进行了证明，实验表明本方案能在多类别图像检索中取得较好的检索效率和检索准确度，且存储成本也是同类对比方案中最小的。同时，针对检索用户权限撤销或图像解密密钥泄露的情况，本文提出了一种支持代理重加密的图像更新算法，该算法由图像所有者生成重加密密钥，委托云服务器对密文图像进行重加密，有效减轻了图像所有者的计算负担。最后，本文对基于深度学习的图像可搜索加密方案中已有类别增加新图像和增加新类别图像两种情况下索引的更新情况进行了讨论和实验。本文的研究内容对于图像可搜索加密算法的研究具有一定的参考意义。

1.2 国内外研究现状

现有的加密图像检索方案大多是基于CBIR框架提出的，根据图像加密和特征提取的顺序，加密图像检索方案可分为基于密文特征的安全图像检索和基于密文图像的安全图像检索两类[5]。基于密文特征的安全图像检索先提取图像特征构建安全索引再加密图像，云服务器利用安全索引进行相似度计算。基于密文图像的安全图像检索则先加密图像，再将密图上传至云服务器，由云服务器完成特征提取和相似度计算等任务。下面将对这两类安全图像检索方案的国内外研究现状进行具体描述。

1.2.1 基于密文特征的安全图像检索

2009年，Lu等人提出了第一个隐私保护的CBIR方案[6]，该方案利用提取的视觉词作为图像特征，并用保序加密和min-hash算法来加密特征。之后，Lu等人在其另一项工作中又提出可通过位平面随机化、随机投影和随机一元编码三种方法对图像特征进行加密，分别支持加密域的汉明距离和L1距离计算[7]。Zhang等人的方案从图像中提取颜色、纹理和形状三种特征，采用Paillier同态加密算法保护图像特征，加密后的特征能直接进行相似度计算[8]。Lu等人的方案中也使用了同态加密算法对特征加密[9]，其方案和Zhang提出的方案均通过同态加密有效保护了图像特征的隐私性，但是同态加密的计算和存储成本往往比较大，且上述两种方案需要频繁与用户交互，会造成较大的通信成本。Wong等人提出了使用安全KNN算法加密图像特征[10]，该算法将密文域索引向量和查询向量的内积比较等同于明文索引向量和查询向量间欧氏距离的比较[11]，该算法在很多方案[12–15]中被使用。Yuan等人提出的SEISA方案中提取图像的Fisher Vector作为特征向量并构建了安全树形检索结构，该方案采用基于轻量级多项式的访问控制策略保证数据所有者可以定义用户的访问权限，此外，该论文中还提出了一种安全的K-means外包算法来节省数据所有者的成本[14]。SEISA提出的多项式访问控制策略在Song等人的方案[16]中也得到了使用。Xia等人提出的基于隐私保护的图像检索方案则利用局部特征SIFT表示图像，并使用线性变换来保护EMD（Earth Mover's Distance）计算中的敏感信息[17]。Yuan等人基于LWE问题[18]提出了安全欧式距离比较算法，并证明了其方案设计在已知密文攻击模型和已知背景攻击模型中的安全性，该算法可被称为基于LWE的安全kNN算法[19]。Shen等人则提出了一种支持多图像所有者的安全CBIR方案[20]，该方案使用了安全多方计算技术来加密图像特征，允许多个图像所有者使用自己的密钥加密图像特征。

随着深度学习的迅速发展，涌现了很多基于卷积神经网络（Convolutional Neural Network，CNN）模型提取特征的安全图像检索方案，与传统的特征相比，

CNN 模型可以更加准确地表示图像内容，使得检索结果也更加准确。2017 年，Li 等人提出的分层加密图像检索方案 CASHEIRS 就采用了深度学习框架 Caffe 来提取图像特征，并通过 PCA-ITQ 技术将提取的 4096 维 CNN 特征降维成低维度的二进制码，在此基础上构建了基于 K-means 聚类算法的安全分层索引结构以提高检索效率[15]。Li 等人则利用 CAK-means 聚类（Combination of AP and K-means clustering）算法[21] 构建索引树，与 K-means 聚类相比，CAK-means 聚类算法使用 AP 聚类初始化 K-means 聚类，使得聚类更加稳定和准确，作者提出的新的安全图像检索方案能够实现高检索准确度和高检索效率[22]。Liu 等人利用深度神经网络模型 MobileNetV2 准确提取图像的关键词集，并通过关键词对图像进行粗粒度的分类以提高检索效率，然后利用图像特征实现细粒度分类，从而检索出准确的结果[23]。之后，还有很多方案为了提高检索的准确性选择使用 CNN 模型提取图像特征[24]，均取得了很好的效果。

此外，许多方案将代理重加密技术应用于图像可搜索加密领域，Li 等人采用基于多项式的访问策略和代理重加密技术实现了支持多用户的加密图像检索方案 CSM[25]，该方案利用代理重加密技术允许多个用户使用不同的私钥加密图像，云服务器作为代理，可以将图像所有者的密文转换为由检索用户各自的私钥解密的密文。之后，Li 等人在其另一项工作中提出了基于边缘计算的支持多密钥的加密图像检索方案[26]，与 CSM 不同，该方案利用代理重加密技术进行密钥转换，以便管理用户访问使用不同加密密钥加密的图像集。与此同时，为了有效阻止检索用户对检索结果进行非法复制和分发，很多方案[13,25,27] 采用水印技术来保护图像的隐私安全。为保证检索结果的正确性，还有很多方案[28] 支持对检索结果的验证。

基于密文特征的安全检索方案能实现较高的检索准确度，但图像所有者需要从明文图像中提取特征向量并构建安全索引，且安全索引还需要一定的存储成本。为了减轻图像所有者的计算负担，研究者们提出了很多基于密文图像的安全图像检索方案。

1.2.2 基于密文图像的安全图像检索

Hsu 等人提出了基于同态加密的安全图像检索方案[29]，该方案使用同态加密算法对图像进行加密，保证云服务器依旧可以从密文图像中提取出原图像的 SIFT（Scale Invariant Feature Transform）特征用于检索。Bai 等人提出的方案[30] 能够从使用同态加密的密文图像中提取 SURF（Speed Up Robust Features）特征用于检索。Bellafqira 等人使用 Pailler 同态加密算法加密图像，并从密图中提取基于小波变换的全局图像特征，该方案不会在客户端和服务器之间引起额外通信[31]。陈帆等人提出了一种结合二进制 SIFT 和同态加密的安全图像检索算法[32]，结合了加密域中 SIFT 特征提取方法及 SIFT 量化。此外，还有很多方案采用同态加密算

法来保护图像隐私安全并实现有效计算。但是，基于同态加密的方案通常会带来很高的计算和存储负担。

Ferreira 等人提出了一个用于大型共享图像存储库中外包隐私保护存储和检索的安全方案 IES-CBIR[33]，作者认为纹理特征比颜色特征更重要，所以选择使用概率加密来加密纹理特征，对颜色特征采用确定性加密。Xia 等人提出了基于加密词袋（bag-of-encrypted-words，BOEW）模型的安全外包 CBIR 方案[34]，该方案通过颜色值替换、块置换和块内像素置换加密图像，云服务器可以从密图中提取局部颜色直方图用于聚类，从而得到聚类中心作为可视词，建立起 BOEW 模型，最后将统计的可视词出现的频率结果归一化作为图像特征向量。Cheng 等人提出了一种基于马尔可夫（Markov）过程的加密 JEPG 图像检索方案，该方案采用流加密和排列加密对 JPEG 图像加密[35]。之后，Cheng 等人又在其另一项工作中采用离散余弦（Discrete Cosine Transform，DCT）系数随机置乱的方法对 JPEG 图像加密，其方案能够在密文图像中保留 DCT 系数的分布信息，通过提取 DCT 系数直方图实现具有隐私保护的相似图像检索[36]。Xia 等人提出了一种应用于工业物联网场景的方案[37]，该方案使用块变换、块内变换和像素替换来加密图像，并使用两个或多个好奇的云服务器协同提取特征，在该方案中，云服务器能够从密文图像中直接提取本地二进制模式（Local Binary Pattern，LBP）直方图来进行检索。之后 Kitayama 等人提出了一种从加密且压缩的图像中提取 HOG 特征的方法，并将提取的 HOG 特征用于机器学习算法[38]。Qin 等人提出了高性能隐私保护 SIFT 特征检测系统 SecSIFT[39]，其工作中采用随机排列的方法来防止形状信息的泄漏，采用 OPE 算法来保证关键点的定位和 SIFT 特征的计算。Jiang 等人提出了在大量加密图像中提取方块截断编码（Block Truncation Coding，BTC）特征的隐私保密方案[40]，在该方案中，所有图像经过位置换、像素扩散和位平面随机加密后能够保持 BTC 特征不变。Chen 等人提出使用块内像素扰乱的算法加密图像并从密文图像中提取图像特征用于检索，但是这种加密算法无法保证图像安全性[41]。

综上所述，基于密文图像的安全图像检索方案将特征提取、索引构建等任务交由云服务器完成，能有效减轻图像所有者的任务，但与基于密文特征的检索方案相比，这类方案的检索准确度往往比较低。

1.3 本文主要研究内容

针对安全图像检索领域检索准确度较低、检索效率较低、存储成本太大等问题，本文对图像特征提取算法、索引构建、特征加密算法和图像加密算法等方面进行了研究，本文的主要研究内容包括：

（1）基于随机矩阵的加密算法

本文提出了一种基于随机矩阵的加密算法应用于图像特征加密，该算法能够实现密文域下余弦相似度的计算，在保证图像隐私安全和检索准确度的同时，该算法使用的特征维度比同类方案更短。

（2）基于深度学习的图像可搜索加密方案

本文通过预训练的 CNN 模型提取图像特征向量，将图像特征向量表示为类别标签和图像特征两部分。利用类别标签构建了基于 K-means 聚类算法的层次索引树以提高检索效率，并使用基于 LWE 的安全 kNN 算法对索引树的节点进行加密。在此基础上，构建了基于深度学习的安全原型图像检索系统，该系统对类别标签和图像特征分别采用基于 LWE 的安全 kNN 算法和基于随机矩阵的加密算法来保证检索索引的安全性，并分析了该系统在已知密文攻击模型和已知背景攻击模型下的安全性，通过实验对比，证明了本方案在多类别图像时能实现高效且高准确的检索，且安全索引的存储开销比同类对比方案小。

（3）图像可搜索加密方案中的更新操作

本文提出了一种支持代理重加密的图像更新算法，当检索用户权限被撤销或图像解密密钥被泄露时，图像所有者重新生成新的公私钥对用于加解密图像，并生成重加密密钥发送给云服务器用于对密文图像重加密，且重加密后的图像只能使用新的私钥解密。同时，本文对基于深度学习的图像可搜索加密方案中已有类别中增加新图像和增加新类别图像两种情况下索引的更新情况进行了讨论。

图1.1给出了本文研究内容的逻辑关系图。

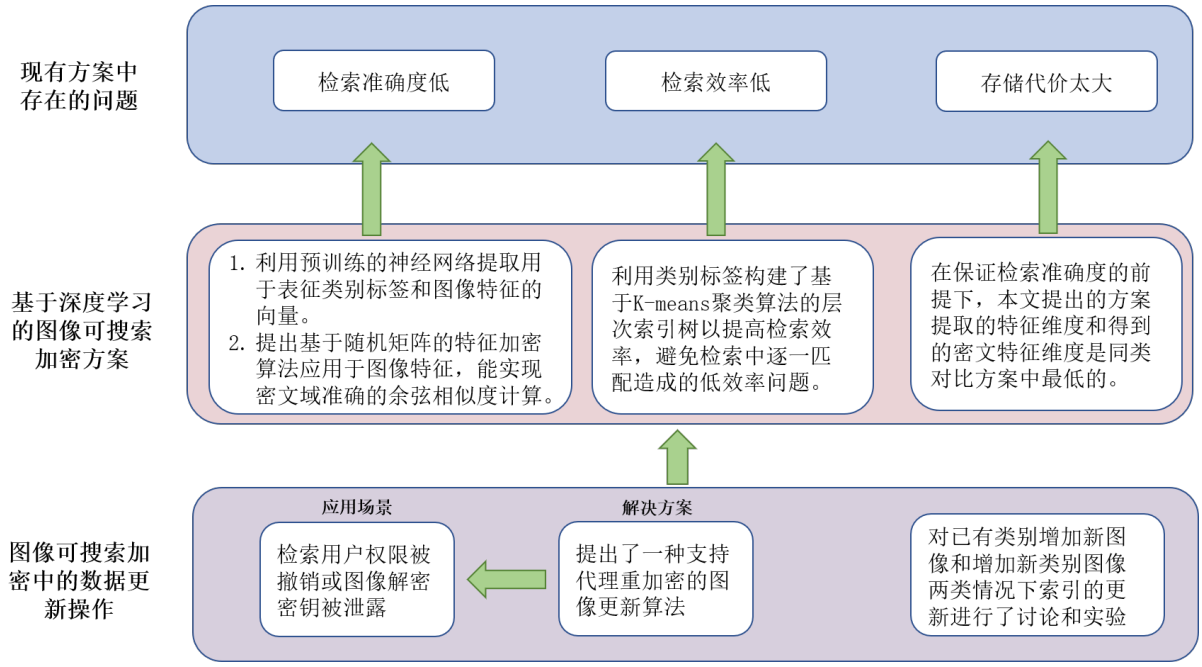


图 **1.1**　本文研究内容逻辑关系图

1.4　本文组织结构

本文对基于深度学习的图像可搜索加密算法进行了研究，主要组织结构如下：

第一章绪论，阐述了图像可搜索加密的研究背景及意义，然后分基于密文特征的安全图像检索和基于密文图像的安全图像检索两个大类介绍了国内外的研究现状。最后给出了本文研究的主要内容，并对文章的组织结构进行了介绍。

第二章相关理论基础，该章节对图像特征的提取、特征降维技术、索引构建方法以及常见的特征和图像加密算法进行了介绍，并给出了图像相似性度量标准以及图像检索性能的衡量标准，最后介绍了本文用到的 K-means 聚类算法。

第三章对本文提出的基于深度学习的图像可搜索加密方案进行了介绍，介绍了该系统的系统框架、威胁模型、图像特征向量提取网络和索引结构等内容，并详细介绍了本文提出的基于随机矩阵的图像特征加密算法，最后通过安全性分析和实验验证证明了方案的可行性。

第四章对图像和索引的更新进行了分析，并提出了支持重加密的图像更新方案，该方案由云服务器对密图进行重加密，使得重加密后的图像只能用新私钥解密。同时，还对已有类别增加新图像和增加新类别图像这两种情况下索引的更新进行了讨论。

第五章总结了本文的工作，并对未来的研究内容进行了展望。

第 2 章　相关理论基础

2.1　图像特征的提取方法

图像特征的提取与表达是基于内容的图像检索的基础，它对图像分类、检索、识别等应用具有十分重要的作用。图像特征提取是将图像转化为一组可供计算机处理的向量或描述符，能够在不同的图像任务中表达出图像的不同属性和特征，从而为后续的图像处理和分析提供基础。下面将从传统特征提取方法和基于深度学习的图像特征提取方法两方面进行说明，并对特征向量降维技术进行介绍。

2.1.1　传统特征提取方法

传统的特征提取方法是基于图像本身特征进行提取，常见的传统特征提取方法包括尺度不变特征变换（SIFT）[42]、加速鲁棒特征（SURF）[43]、局部二值模式（LBP）[44,45]、方向梯度直方图（Histogram of Oriented Gradient，HOG）[46] 等。

SIFT 算法用于提取图像中关键的局部特征，且对旋转、缩放、亮度等因素具有较强的不变性。SIFT 特征的提取过程主要包括四步：（1）通过高斯差分金字塔检测出图像在不同尺度和位置上的极值点；（2）对尺寸空间中的极值点进行精确的位置定位，得到关键点；（3）为每个关键点分配主方向，使其具有旋转不变性；（4）根据关键点周围的图像梯度生成局部特征向量。SIFT 算法提取的特征稳定，但是计算量大，对于边缘光滑目标的特征点提取比较弱。

SURF 是对 SIFT 算法的一种改进，与 SIFT 算法相比，SURF 算法的计算速度更快、尺度不变性更好。SURF 算法通过使用快速的 Haar 小波响应和加速积分图技术，实现了计算速度和特征提取性能之间的平衡。

LBP 是一种局部纹理特征描述子，其算法简单、计算量较小。原始的 LBP 算法的主要步骤可以概括为四步：（1）选择一个像素周围的邻域，例如 8 邻域或者 16 邻域。（2）将邻域中的像素值与中心像素值进行比较，若邻域的像素值大于中心像素值，则该邻域像素值标记为 1，否则标记为 0。将邻域的二进制值组成二进制编码，即 LBP 特征。（3）对图像中的每个像素计算 LBP 编码，得到 LBP 模式直方图，即 LBP 特征向量。如图2.1所示，我们选择了一个像素值的 8 邻域，计算得到其 LBP 特征为 $(00101110)_2$=46。

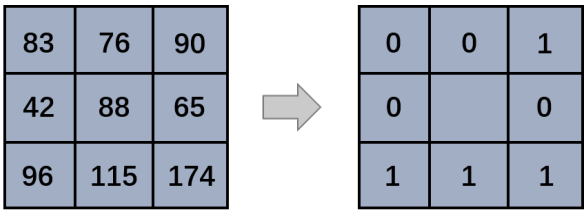


图 **2.1**　**LBP** 特征计算示例

除上述的三种传统特征提取算法，HOG 也是一种常见的图像特征提取算法，该算法主要用于目标检测和行人识别等计算机视觉领域的任务。HOG 算法通过计算图像中不同方向的梯度直方图来描述图像的局部特征，进而构建出整个图像的特征向量。HOG 算法主要分为以下几个步骤：（1）图像预处理：将图像转化为灰度图像，并进行归一化和平滑处理；（2）计算梯度：通过使用算法（如 Sobel 算子）来计算图像中每个像素点的梯度，得到梯度的幅值和方向；（3）图像划分：将图像划分为 cell，每个 cell 内包含多个像素点；（4）统计每个 cell 的像素点，将它们的梯度方向统计到一个直方图中，可以将直方图分为若干个 bin，每个 bin 对应一定范围的梯度方向；（5）归一化：对每个 cell 内的梯度直方图进行归一化，使得它们在不同的光照条件下也能比较；（6）组合相邻的 cell：将临近的 cell 合并成一个更大的 cell 以得到更具有鲁棒性的特征描述；（7）得到 HOG 特征向量：将所有 cell 的特征向量按照一定顺序组合起来得到 HOG 特征向量。

2.1.2 基于深度学习的特征提取方法

深度学习的特征提取基于数据本身的分类特性。深度学习的特征提取步骤通常可分为三步：构建数据集、搭建深度学习模型、训练。

Yann LeCun 在 1998 年提出第一个成功的卷积神经网络模型 LeNet[47]，该模型可用于识别手写数字。LeNet 使用卷积层、池化层等基本神经网络组件，这些组件后来被广泛应用于各种深度学习模型中。Alex Krizhevsky 在 2012 年提出首个大规模应用卷积神经网络的模型 AlexNet[48]，该模型在 ImageNet 大规模图像识别挑战赛中获得了顶级表现。AlexNet 总共 8 层网络，其中 5 层是卷积层，3 层是全连接层，该模型的输入图像大小为 224×224，输出层包括 1000 个神经元，分别对应 ImageNet 数据集中的 1000 个类别。AlexNet 在计算机视觉的成功推动了深度学习的发展，激励了更多的研究人员去探索更深、更高效的神经网络模型。2014 年，Oxford 的 Visual Geometry Group 提出了 VGG（Visual Geometry Group）[49]，VGG 网络的特点是网络结构非常简单和规整，由多个相同大小的卷积层和池化层构成，容易优化。但是由于其层数过多，参数量巨大，计算量较大，需要更长的训练时间和更大的存储空间。

2015 年，微软研究院提出的 ResNet（Residual Network）[50]，通过使用残差连接来解决深度卷积神经网络中梯度消失和梯度爆炸的问题。ResNet 中跳跃连接的作用是直接将之前层的特征图传递给后续层，以便更好地传递低层次的信息，避免信息的逐层丢失。如图2.2所示，ResNet 网络与传统的前馈网络相比，增加了跨层连接，ResNet 中 l 层的输出是 $l-1$ 层的输出加上对 $l-1$ 层输出的非线性变换，即 $x_l = H_l(x_{l-1}) + x_{l-1}$，由于跳跃连接不涉及到任何参数的学习，因此其不会增加网络的参数数量或计算负担，反而能够增强网络深度，提高网络的性能，

ResNet 的提出也为后来的深度网络模型的设计提供了重要的参考和启示。2017 年，Gao Huang 等人提出了 DenseNet（Dense Convolutional Network）[51]，与之前提出的网络不同，该网络采用的是密集连接，即每个层的输出与前面所有层的输出相连，对于 DenseNet 网络的第 l 层，将之前的所有层 $[x_0, x_1, ..., x_{l-1}]$ 拼接作为输入，即 $x_l = H_l([x_0, x_1, ..., x_{l-1}])$，共包括 $\frac{l(l+1)}{2}$ 个连接。其中，$H_l(*)$ 表示非线性转化函数，它是一个组合操作，可能包括一系列的 BN（Batch Normalization）、ReLU、Pooling 和 Conv 操作[51]。这种设计使得 DenseNet 可以在更少参数数量和计算量的情况下实现更高的准确率，因此被广泛应用于图像识别、语音识别等任务中。

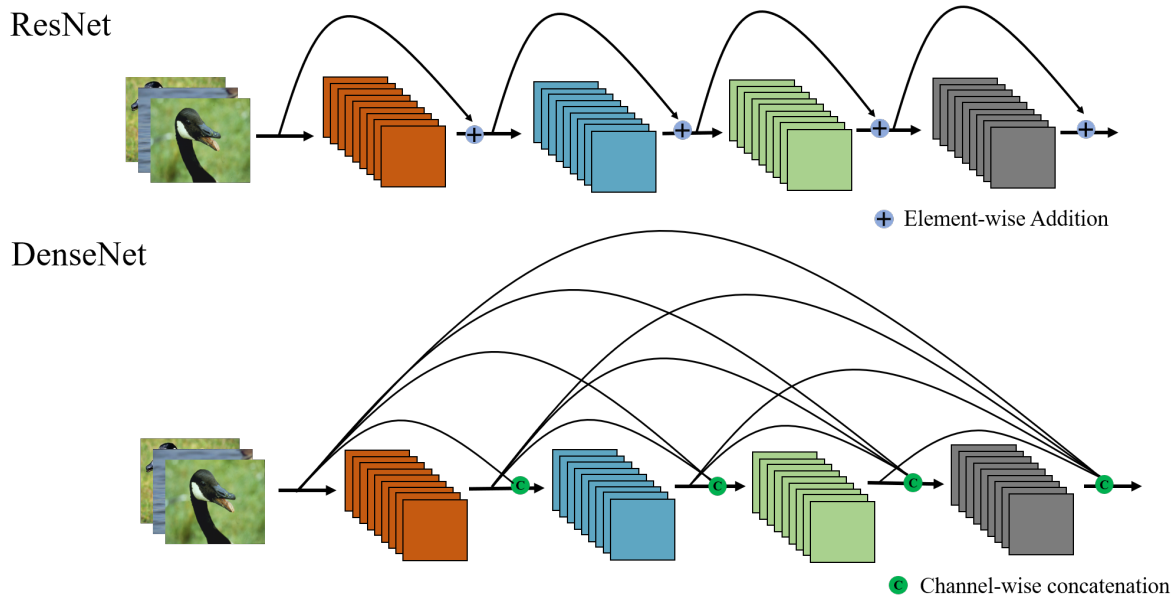


图 **2.2**　**DenseNet** 网络与 **ResNet** 网络的连接机制

图2.3描述的是一个包含三个 Dense Block 的 DenseNet 网络结构，Dense Block 中的非线性组合函数使用的是 BN+ReLU+3×3 Conv 的结构，由于 DenseNet 特征重用，导致后面层的输入非常多，针对这一问题，DenseNet 采用 bottleneck 层来减少计算量，即 BN+ReLU+1×1 Conv+BN+ReLU+3×3 Conv 结构。同时，在两个相邻的 DenseBlock 间，使用 Transition 层连接，以降低特征图的大小。Transition 层结构为 BN+ReLU+1×1 Conv + 2×2 AvgPooling。

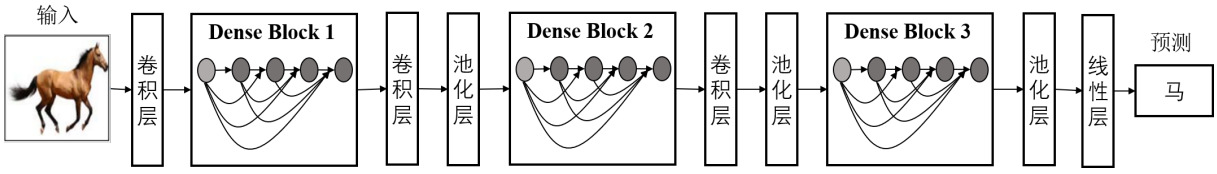


图 **2.3**　**DenseNet** 网络结构[51]

近些年也出现了将深度学习与哈希技术相结合的方法，比较经典的有卷积网络哈希（Convolutional Neural Network Hashing，CNNH）。Lu 等人提出的方法[52] 中利用目标哈希码作为训练标签，假设有 K 类训练集图片，该方案通过贪心法生成

*K*bits 的二进制目标哈希码，使得二进制码组中两两之间的汉明距离可以达到最大。Lin 等人提出了一种有效的深度学习框架，以生成二进制哈希码并实现快速图像检索[53]。这个框架对网络模型进行了修改，在最后一层之前添加了一个潜在层（全连接层），用于学习表示类别标签的二进制编码。通过添加这个潜在层，该方案可以同时学习图像表述和二进制编码。此外，该方案还提出了分层深度搜索结构，即利用二进制编码进行粗粒度的检索，再利用图像自身的特征进行细粒度检索，该方案在大规模数据集上进行了应用。

与传统的特征提取方法相比，基于深度学习的特征提取方法能够自动学习特征，对于图像变形、光照、噪声等问题的鲁棒性更好，且在大规模数据集上能实现很好的准确率。但是这种方式也存在一些缺点：（1）训练成本高：深度学习需要大量的数据进行训练，对于 GPU 等资源有一定的要求，并且需要一定时间和人力对网络结构进行调整。（2）模型的可解释性较差：深度学习的模型往往具有很高的复杂度，很难解释模型如何得出最终的特征表示。

因此，对于图像特征提取方法的选择应该针对具体问题和应用场景进行选择。

2.1.3 主成分分析

在计算机视觉和机器学习中，图像特征通常是高维的向量，这会带来计算和存储方面的问题，因此，需要对图像特征进行降维处理。特征向量经过降维可以提高算法的计算效率、减少存储空间。主成分分析（Principal Component Analysis，PCA）是常见的一种降维技术，也是在本文中使用到的一种降维技术，下面将对 PCA 进行详细介绍。

PCA 技术的核心思想是将数据投影到新的坐标系上，使得投影后的数据在新坐标系下方差最大，这些新的坐标轴被称为主成分，每个主成分都是原始特征的线性组合。通过保留前几个主成分，可以实现对数据的降维，同时尽可能地保留原始数据的信息。

以下是 PCA 算法的基本步骤：

（1）数据中心化：将每个特征的平均值移动到 0，以消除特征间的偏差。

（2）计算协方差矩阵：计算每对特征之间的协方差。

（3）计算特征值和特征向量：对协方差矩阵进行特征分解，得到特征值和对应的特征向量。

（4）选择主成分：将特征向量按照对应的特征值大小排序，选取前 k 个特征向量作为主成分。

（5）投影数据：将数据投影到选择的主成分上，得到降维后的数据。

2.2 索引构建方法

（1）线性检索结构

在线性检索时，云服务会将用户提交的查询向量与特征库中的特征逐一匹配，并计算它们之间的相似度。Li 等人在[54] 中提出的方案采用了线性检索结构。当需要检索的数据量较小且数据数量固定时，线性检索是一种简单且直观的检索方式。然而，随着数据量的增加，线性检索的效率会逐渐降低。

（2）双层检索结构

常见的双层检索结构是基于局部敏感哈希（Locality Sensitive Hashing，LSH）的双层检索。LSH 算法是一种针对海量高维数据的快速最近邻查找算法，其基本思想可以概括为：原始数据空间中的两个相邻的数据点通过相同的 LSH 函数进行映射或者投影变换后，被映射到同一个桶的概率较大，而不相邻的数据点被映射到同一个桶的概率较小[11]。LSH 的双层索引结构在很多方案[13,17,55] 中被应用。Xia 等人的方案[13] 中使用 LSH 构建了 Two Layer Index 双层索引结构，索引顶层是利用 p 稳定分布的 LSH 构建的散列表，用于过滤图像以减小搜索范围，底层是一对一的索引，用于检索时的相似度计算。与线性检索相比，双层检索结构的检索效率更高。

（3）树形检索结构

除上述两种索引结构外，树形检索结构也是 CBIR 方案中常见的检索结构之一。其中 KD-Tree（k-dimensional Tree）是具有代表性的一种算法。之后，很多方案将聚类算法与树形检索结构融合，如：Yuan 等人提出使用 K-means 聚类算法将图像划分为多个类，相似的图像会被划分到同一类，之后分别在各类中执行 K-means 聚类算法，直到每类中图像数量均少于 T 则完成索引树构建[14]。Li 等人提出的分层索引树使用 K-means 聚类的结果作为叶节点构建索引树[15]，缩小了搜索范围，避免在搜索阶段遍历整个图像集。Li 等人采用 CAK-means 构建树形检索结构[22]，CAK-means 使用 AP 聚类的结果对 K-means 聚类进行初始化，解决了 K-means 聚类算法对初始值敏感的问题。上述树形检索结构的使用有效的提高检索效率，但与线性检索结构和双层检索结构相比，构建树形检索结构的时间开销较大，同时，如何设计高效平衡且检索准确度较高的树形检索结构是一个关键的问题。

（4）倒排索引

倒排索引最初用于文本文档，在文本检索中，可以利用倒排索引快速定位到包含关键词的文档，在检索过程中，只需要考虑出现在查询词倒排索引中的那些文档。利用图像的视觉词表示，倒排索引技术可以扩展到图像检索领域。在图像检索中，倒排索引常和基于视觉词袋模型（Bag of Word，BOW）一起使用，BOW

借鉴文本检索的思路，从图像中提取局部特征（如 SIFT），再利用聚类算法（如 K-means）构建出视觉单词。对于图像中的每一个局部特征，都能从字典中找到最相似的视觉词，通过统计该图像的局部特征在字典中的视觉词出现的频率，得到图像的直方图向量作为图像的特征向量。最后通过 TF-IDF（Term Frequency–Inverse Document Frequency）过滤不重要的词汇，构建倒排索引。Lu 等人将倒排索引应用于加密域图像检索方案中，其方案采用保序加密算法对词频信息加密，保护了词频信息的安全[7]。

2.3 图像和特征的加密技术

2.3.1 同态加密

同态加密（Homomorphic Encryption，HE）是一种在密文域上支持加法、乘法等基本操作的加密算法。同态加密允许在不解密的状态下对密文进行计算得到密文结果，并且将密文结果进行解密得到的结果与用同一方法处理明文数据得到的输出是一样的[56]。这种技术可以保护数据的隐私性，并使得数据共享和处理更加方便。同态加密可分为部分同态加密（Partially Homomorphic Encryption，PHE）和全同态加密（Fully Homomorphic Encryption，FHE）两类。

在部分同态加密中，原始数据被加密成密文，可以在不解密的情况下进行加法或者乘法运算。例如：假设有两个密文数字 $E(a)$ 和 $E(b)$，若想要计算 $a+b$，对于传统的加密算法，需要解密后获得 a 和 b 才能得到结果。但是在部分同态加密中，可以直接对这两个数字密文进行加法运算获得加密和，如公式（2.1）所示。

$$E(a+b)=E(a)\oplus E(b) \tag{2.1}$$

对得到的结果进行解密可以获得如公式（2.2）中的明文。

$$a+b=D(E(a+b)) \tag{2.2}$$

其中 $E(*)$ 和 $D(*)$ 分别代表加密运算和解密运算。乘法同态加密与加法同态加密类似，能够实现密文下的乘法运算。部分同态加密的实现方式有很多种，如 RSA 同态加密、Paillier 同态加密和 ElGamal 同态加密等，这些方法都使用了不同的数学方法来实现部分同态加密，其中 Paillier 同态加密是应用最广的，其支持加法同态加密。有限同态加密则允许一定程度的加法和乘法计算，但是其只能支持有限的计算操作。Boneh 等人使用类似于 Paillier 的构造方法和双线性映射构造了一种同态加密方案[57]，该方案支持对加密数据进行任意加法和一次乘法。

全同态加密最早是由 Craig Gentry 在 2009 年提出的[58]，其被誉为密码学领域

的重大突破之一。全同态加密可以在不解密的情况下进行任何计算，包括加法、乘法、比较等。与部分同态加密只支持有限的计算不同，全同态加密支持对密文进行无限制的计算。全同态加密的实现方法有很多种，包括基于格的全同态加密、基于 RLWE（Ring Learning With Errors）的全同态加密等等。这些方法都使用了不同的数学方法来实现全同态加密。

在图像可搜索加密领域，同态加密算法可被用于加密图像或者加密图像特征向量，利用该算法支持密文计算的特点来实现密文域上的相似度计算。Lu[9] 和 Zhang[59] 等人将同态加密技术应用于特征向量加密，加密后的密文向量能直接进行相似度计算。Hsu[29] 和 Bai[30] 等人使用同态加密算法加密原始图像，云服务器依旧可以从密图中提取出原图像的特征用于检索。

然而同态加密算法需要对数据进行复杂的计算，其时间复杂度比传统的加密算法更高，这可能会导致其在实际应用中的局限性。

2.3.2 保序加密

保序加密（Order-Preserving Symmetric Encryption，OPE）是 Agrawal 等人[60] 在 2004 年提出的一种加密技术，该加密技术允许在加密后仍然能够对数据进行排序和比较，而不需要解密数据，可用于需要保护隐私的数据排序和搜索应用程序，OPE 可表示为公式（2.3）的形式。

$$c_i = f_{OPE}(m_i) \tag{2.3}$$

其中 m_i 表示明文，c_i 表示经过 OPE 加密的密文，若 $m_1 < m_2$，则 $c_1 < c_2$。2009 年，基于超几何分布的伪随机对称保序加密算法 SE[61] 在 2009 年被提出，该算法首次证明 OPE 能够抵御弱选择明文攻击。在 Xia 等人[37] 提出的方案中，采用[61] 提出的 OPE 生成保序映射来加密图像的像素值。保序加密技术的突出特点是密文数据能够直接进行排序，基本无需用户参与交互，有效降低了密文搜索的难度，但保序加密在实际应用中存在较大的安全风险，特别缺乏理论层面的安全性支持[62]。

2.3.3 安全 KNN 算法

Wong 等人[10] 在 2009 年提出了安全 KNN 加密算法加密图像特征，使得加密的特征向量能够计算欧式距离。这种算法在很多方案[12,25] 中被使用，下面对该算法进行详细介绍。我们首先假设图像特征向量 $p = (p_1, p_2, ..., p_d)^T$，查询特征向量为 $q = (q_1, q_2, .., q_d)^T$，该算法的主要步骤可概括为以下四步：

（1）密钥生成：图像所有者生成的用于加密图像特征向量的密钥包括：两个 $(d+1) \times (d+1)$ 维的随机可逆矩阵 M_1 和 M_2，$(d+1)$bits 的随机二进制向量 S。

（2）特征加密：对于图像特征集中的特征向量 p，图像所有者先将 d 维的特征向量扩展为 $d+1$ 维的 $p^{'}=(p_1,p_2,...,\|p\|^2)^T$，再利用随机向量 S 将 $p^{'}$ 拆分为两个向量 $\{p_a^{'},p_b^{'}\}$，具体拆分方式如公式（2.4）所示。最后，图像所有者利用随机矩阵 M_1 和 M_2 加密 $p_a^{'}$ 和 $p_b^{'}$，得到加密的特征向量 $p^{''}=\{M_1^Tp_a^{'},M_2^Tp_b^{'}\}$。

$$\begin{cases} p_a^{'}[j]=p_b^{'}[j]=p^{'}[j], & if\ S[j]=0 \\ p_a^{'}[j]+p_b^{'}[j]=p^{'}[j], & if\ S[j]=1 \end{cases} \tag{2.4}$$

（3）生成查询陷门：对于查询向量 q，检索用户首先也将其扩展为 $(d+1)$ 维向量 $q^{'}=(-2q_1,-2q_2,...,-2q_d,1)^T$。之后，按照公式（2.5）所示的方法，检索用户将 $q^{'}$ 拆分为 $q_a^{'}$ 和 $q_b^{'}$ 两部分。最后将 $q_a^{'}$ 和 $q_b^{'}$ 加密为 $q^{''}=\{\gamma M_1^{-1}q_a^{'},\gamma M_2^{-1}q_b^{'}\}$，其中 γ 为随机正整数。

$$\begin{cases} q_a^{'}[j]+q_b^{'}[j]=q^{'}[j], & if\ S[j]=0 \\ q_a^{'}[j]=q_b^{'}[j]=q^{'}[j], & if\ S[j]=1 \end{cases} \tag{2.5}$$

（4）相似度计算：如公式（2.6）云服务器收到检索请求后，利用密文索引和查询陷门进行相似度计算，将相似度最高的 k 张图像返回给检索用户。

$$\begin{aligned} Sim &= (\gamma M_1^{-1}q_a^{'})^TM_1^Tp_a^{'}+(\gamma M_2^{-1}q_b^{'})^TM_2^Tp_b^{'} \\ &= \gamma p^{'T}q^{'} \\ &= \gamma(\sum_{i=1}^{d}p_i^2-2\sum_{i=1}^{d}p_iq_i) \\ &= \gamma(\|p-q\|^2-\|q\|^2) \end{aligned} \tag{2.6}$$

通过上述安全 KNN 算法加密特征向量后，云服务器可以利用密文特征向量计算得到 p 和 q 之间的欧式距离，但是上述安全 CNN 算法被证明无法抵御线性分析攻击。后续提出了增强的安全 KNN 算法——基于误差学习（learning with errors, LWE）[18] 的安全 kNN 算法，该方案得到了广泛的应用[54,63]。为保护类别标签隐私安全，本文采用了该算法加密类别标签，在第三章中将对该算法进行详细介绍。

2.4 图像相似度度量标准和性能衡量指标

2.4.1 图像相似度度量标准

在图像检索中，需要根据相似度度量标准衡量图像之间的相似性，下面我们简单介绍了几种常见的图像相似度度量标准。

（1）欧式距离（Euclidean Distance）

欧式距离是常见的距离度量之一，其用于衡量N维空间两点之间的真实距离。假设有两个向量 X 和 Y，其中 $X=(x_1,x_2,...,x_N)$，$Y=(y_1,y_2,...,y_N)$，则它们之间的欧式距离 $d(X,Y)$ 可被定义为公式（2.7）中的形式。

$$d(X,Y)=\sqrt{(x_1-y_1)^2+(x_2-y_2)^2+...+(x_N-y_N)^2} \tag{2.7}$$

（2）汉明距离（Hamming Distance）

汉明距离在信息理论中表示两个等长二进制字符串对应位置上不同字符的数量。我们用 $H(X,Y)$ 来表示两个向量 $X=(x_1,x_2,...,x_N)$ 和 $Y=(y_1,y_2,...,y_N)$ 的汉明距离，其计算方法如公式（2.8）所示。

$$H(X,Y)=\sum_{i=1}^{N}|x_i-y_i|,x_i,y_i\in\{0,1\} \tag{2.8}$$

（3）余弦相似度（Cosine Similarity）

余弦相似度用一个向量空间中两个向量夹角间的余弦值作为衡量两个个体之间差异的大小。两个向量 $X=(x_1,x_2,...,x_N)$ 和 $Y=(y_1,y_2,...,y_N)$ 的余弦相似度 $C(X,Y)$ 可以通过公式（2.9）计算得到。

$$C(X,Y)=\frac{\sum_{i=1}^{N}(x_i\times y_i)}{\sqrt{\sum_{i=1}^{N}{x_i}^2}\times\sqrt{\sum_{i=1}^{N}{y_i}^2}} \tag{2.9}$$

2.4.2 性能衡量标准

检索性能的衡量主要考虑检索准确度、检索效率和信息安全三个方面。

（1）检索准确度：通常用查全率（Recall，R）和查准率（Precision，P）来衡量检索是否准确。表2.1中给出了计算查全率和准确度的相关参数。

表 **2.1** 查全率和查准率相关参数

标签	预测	结果
Positive	Positive	True Positive(TP)
Positive	Negative	False Negative(FN)
Negative	Positive	False Positive(FP)
Negative	Negative	True Negative(TN)

查准率（精度）的计算方式如公式（2.10）所示。

$$Precision=\frac{TP}{TP+FP} \tag{2.10}$$

查全率（召回率）的计算方式如公式（2.11）所示。

$$Recall = \frac{TP}{TP + FN} \tag{2.11}$$

（2）检索效率：检索的效率直接关系到用户的检索体验，检索效率通常和特征维度、加密算法、检索结构、硬件设备等相关。

（3）信息安全：图像外包存储需要考虑外包信息的安全性，在图像可搜索加密中，信息的安全性主要包括图像内容安全、索引和查询的安全。

2.5 K-means 聚类算法

K-means 聚类算法是一种常用的无监督机器学习算法，该算法的思想是随机选择 K 个对象作为初始的聚类中心，计算每个对象到各个聚类中心的距离，将每个对象分配给距离它最近的聚类中心。等待第一轮聚类完成后，对每个簇的聚类中心进行更新。进行多次迭代，直至聚类中心不再改变，聚类完成。与其他聚类算法相比，K-means 聚类算法的时间复杂度较低，为 $O(nmkt)$，其中 n 为数据集中数据的数量，m 表示数据维度，t 表示算法迭代的次数，K 表示簇的数量，K-means 算法在图像分割、文本聚类和市场细分等方向得到了广泛的应用。下面我们将给出 K-means 聚类算法的具体实现。

对于样本集 $X = \{X_1, X_2, ..., X_n\}$，其中 n 表示样本数量，每个样本的维度为 d，K-means 的具体实现可划分为以下几个步骤：

（1）初始化：选择 K 个随机样本 $C_1, C_2, ..., C_K$ 作为初始簇中心。

（2）分配数据点：通过计算每个样本到聚类中心的距离，将每个样本分配到与其计算距离最近的簇中心，具体计算方法如公式（2.12）所示。

$$D(X_i, C_j) = \sqrt{\sum_{p=1}^{d}(X_{i,p} - C_{j,p})^2} \tag{2.12}$$

其中，X_i 代表样本集中的第 i 个样本，$1 \leqslant i \leqslant n$，$C_j$ 表示第 j 个聚类中心，且 $1 \leqslant j \leqslant K$。

（3）更新簇中心：对于每个簇，计算所有样本的平均值来确定新的簇中心。

（4）重复步骤 2 和步骤 3，直到簇中心不再改变或达到预定的迭代次数，聚类完成。

此外，在 K-means 算法中，簇中心的选择对于聚类结果很重要，因此，如果簇中心的选择不合适，聚类结果可能不准确。

2.6 本章小结

本章节从传统特征提取方法和基于深度学习的特征提取方法两个角度对现有的特征提取方法进行了详细介绍，并对特征提取后常用的降维技术 PCA 进行了叙述。之后，结合相关方案对几种常见的检索结构进行了介绍并详述了同态加密、保序加密以及安全 KNN 算法在加密图像检索领域的应用。为了更好的评估方案的性能，还给出了图像相似度度量标准和检索性能评估指标。最后，对本文方案中采用的 K-means 聚类算法进行了介绍。

第 3 章　基于深度学习的图像可搜索加密方案

3.1　引言

本章节主要介绍了一种基于深度学习的图像可搜索加密方案。该方案采用CNN 模型提取两段式图像特征向量，使得每张图像的特征向量可表示为两部分：类别标签（粗粒度检索）和图像特征（细粒度检索）。利用类别标签构建了基于K-means 聚类算法的层次索引树以提高检索效率，并采用基于 LWE 的安全 kNN 算法对索引节点加密，对图像特征则采用本文提出的基于随机矩阵的加密算法，两种加密算法分别实现了密文域欧式距离和密文域余弦相似度计算，保护图像特征向量的同时能提高检索准确度。最后，对方案的安全性进行了详细分析，通过在现有图像集 Caltech256 上进行实验并与同类方案进行对比，证明本文提出的方案能够实现多类别图像高效且准确的检索，且存储代价也是同类方案中最小的。

3.2　问题阐述

3.2.1　系统模型

本文提出的方案主要系统模型如图3.1所示。系统主要包含三个实体：图像所有者（Image owner）、检索用户（Search user）和云服务器（Cloud server），各实体的主要任务如下：

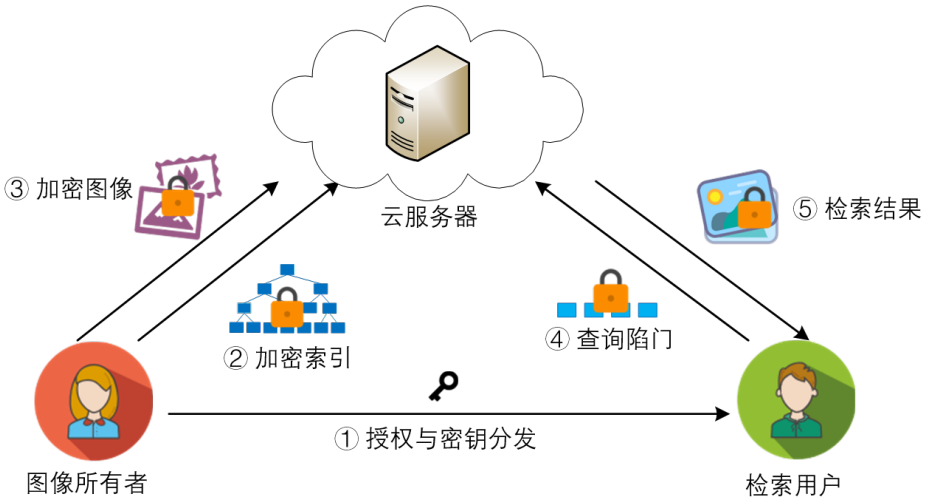


图 **3.1**　系统模型

（1）图像所有者：图像所有者拥有图像集 $\mathcal{M}=\{m_1,m_2,...,m_n\}$，$n$ 表示图像的数量。图像所有者首先从明文图像中提取特征向量并构建明文索引 $\mathcal{I}$，然后对索引 $\mathcal{I}$ 和图像集 $\mathcal{M}$ 进行加密获得安全索引 $\widetilde{\mathcal{I}}$ 和密文图像集 $\mathcal{C}$，最后将安全索引

和密文图像集上传到云服务器中存储。此外，图像所有者还需要向授权用户分发密钥用于生成检索陷门和解密图像。

（2）检索用户：检索用户在进行检索时，首先利用预训练的模型提取图像特征向量，并使用密钥生成查询陷门，将查询陷门提交至云服务器进行检索。当收到来自云服务器的检索结果时对其解密获得明文图像。

（3）云服务器：云服务器主要提供存储和计算服务，存储密文图像和安全索引，在收到查询请求时进行相似度计算并返回检索结果给检索用户。

系统中的算法主要分为六个部分，分别是：生成密钥（GenKey）、加密图像（EncImage）、生成安全索引（GenIndex）、生成查询（GenQuery）、检索（Search）和解密图像（DecImage），下面对各算法进行简要介绍。

（1）GenKey：图像所有者生成密钥集 ψ_O 和 ψ_U，其中密钥 ψ_O 用于图像所有者加密图像和生成安全索引，ψ_U 分发给授权用户生成查询陷门和解密检索结果。

（2）EncImage：图像所有者对图像集 $\mathcal{M}$ 中的每张图像 m_i 使用密钥 pk 加密获得密文图像集 $\mathcal{C}$。

（3）GenIndex：GenIndex 算法又可进一步划分以下三部分：

① 特征提取：图像所有者利用预训练的 CNN 模型提取图像特征向量，得到特征向量集合 $\mathcal{H}$。

② 生成索引：使用图像特征向量构建明文索引 $\mathcal{I}$。

③ 索引加密：图像所有者使用加密算法对明文索引 $\mathcal{I}$ 加密得到安全索引 $\widetilde{\mathcal{I}}$。

（4）GenQuery：检索用户在提交查询前，需要利用预训练的 CNN 模型提取检索图像的特征向量 H_q，然后使用密钥生成查询陷门并上传到云服务器进行检索。

（5）Search：云服务器收到来自检索用户的检索请求时，进行相似度计算，得到相似度最高的 k 张图像的集合 $\mathcal{R}'$ 返回给检索用户。

（6）DecImage：检索用户对检索结果 $\mathcal{R}'$ 解密后获得明文检索结果 $\mathcal{R}$。

3.2.2 威胁模型

本方案中认为图像所有者是完全可信的，其不会与云服务器或其他非法用户勾结。云服务器是诚实且好奇的，能够诚实的执行协议，但是会通过分析数据和安全索引来获得与明文图像的相关信息。基于上述情况，本方案主要考虑以下两类威胁模型：

已知密文攻击模型：在已知密文攻击模型中，云服务器已知图像所有者外包的密文图像和安全索引、检索用户上传的陷门、以及返回的检索结果。

已知背景攻击模型：在已知背景模型攻击中，云服务器除了能知道已知密文攻击中的所有信息，它还能根据查询中的统计信息推断出特殊信息，但是它不知道查询的明文信息，即，云服务器无法获得明文和密文对。

3.2.3 设计目标

为了实现抵御上述攻击模型的安全图像检索方案，本文方案必须满足以下 4 个方面的安全性和性能要求。

（1）检索准确度：设计的方案应选择合适的加密算法和相似度衡量指标，提供尽可能准确的查询结果以满足用户需求。

（2）检索效率：方案应设计合适的检索结构和加密算法提高检索效率，以适用于云服务器中海量数据的检索。

（3）存储开销：方案生成的安全索引和密文图像所占的存储空间应尽可能小。

（4）隐私安全：本方案中的隐私安全主要包括：① 图像安全；② 索引安全与查询隐私；③ 查询陷门的不可链接性。

① 图像安全：对于加密后的图像，只有拥有解密密钥的合法授权用户才能解密，其他没有密钥的用户均无法从密文图像中获取明文信息。

② 索引安全和查询隐私：索引（包括索引节点和图像特征）和查询请求的隐私都应该得到保护，防止云服务器从密文索引和查询陷门中获得明文向量。

③ 查询陷门的不可链接性：每次检索时，云服务器都已知安全索引和查询陷门，为了保护隐私性，即使是相同的明文，每次查询生成的陷门也应不同的，使得云服务器无法推断出多个查询之间的关系。

3.2.4 符号定义

本小节在表3.1中给出了本方案中用到的参数具体的定义，并给出了本章节中用到的三种数学运算的定义。

表 **3.1** 变量及描述

变量	定义
$\mathcal{M}=\{m_1,m_2,...,m_n\}$	明文图像集
$\mathcal{C}=\{c_1,c_2,...,c_n\}$	密文图像集
$\mathcal{H}=\{H_1,H_2,...,H_n\}$	图像特征向量集合
$H_i=(H_i^{(1)}\|\|H_i^{(2)})=(h_{i,1},h_{i,2},...,h_{i,d_1},f_{i,1},f_{i,2},...,f_{i,d_2})$	第 i 张图像的特征向量
$H_q=(H_q^{(1)}\|\|H_q^{(2)})=(h_{q,1},h_{q,2},...,h_{q,d_1},f_{q,1},f_{q,2},...,f_{q,d_2})$	查询图像的特征向量
d_1,d_2	类别标签维度，图像特征维度
d	图像特征转换后的维度
$ck=\{ck_1,ck_2,..,ck_K\}$	K-means 聚类中心
$\mathcal{I},\widetilde{\mathcal{I}}$	明文索引，安全索引
$\mathcal{R},\mathcal{R}'$	明文检索结果，密文检索结果

$\lceil x\rfloor$ 表示最接近 x 的整数，$\lceil x\rfloor_p$ 表示 $\lceil x\rfloor \, mod \, p$。对于一个矩阵 M，$tr(M)$ 表示矩阵 M 的迹。对于向量 $\vec{\xi}$，$max|\vec{\xi}|$ 表示向量中元素最大值的绝对值。

3.2.5 基于深度学习的特征向量提取模型

本文采用两段式特征来表示图像的特征向量。先将设计好的目标哈希码放入改进的网络中训练得到预训练的 CNN 模型，具体的 CNN 模型结构如图3.2所示，该 CNN 模型使用 DenseNet 网络作为基础模型，使用设计的目标哈希码代替网络训练中的 one-hot 标签，并使用交叉熵函数来计算实际输出与目标哈希码的损失值，使得输出的图像类别标签逼近目标哈希码。每张图像使用预训练的 CNN 模型提取的图像特征向量可表示为类别标签 $H^{(1)}$ 和图像特征 $H^{(2)}$ 两部分，类别标签和图像特征分别用于粗粒度检索和细粒度检索。

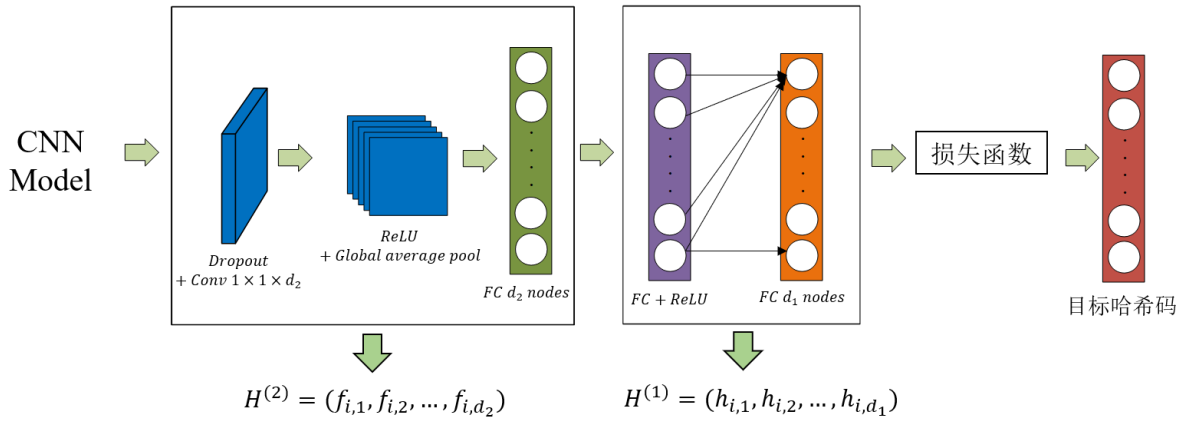


图 **3.2** 特征提取网络模型

为使目标哈希码保持图像类别之间的相关性，设计的目标哈希码应满足：若 $d(f_s, f_{t_1}) < d(f_s, f_{t_2})$，则表明 f_s 和 f_{t_1} 比 f_s 和 f_{t_2} 更相似，其中 $d(*,*)$ 表示两个向量的汉明距离。例如，有猫、狗和海洋三类图像，三类图像中显然猫和狗的相似度比较高，因此猫和狗的类别标签的汉明距离小于猫和海洋、狗和海洋类别标签的汉明距离，即满足：$d(f_{cat}, f_{dog}) < d(f_{cat}, f_{ocean})$ 且 $d(f_{cat}, f_{dog}) < d(f_{dog}, f_{ocean})$，如表3.2所示。

表 **3.2** 目标哈希码举例

图像类别	目标哈希码
猫	11101110
狗	11011110
海洋	00010101

为了得到上述要求的目标哈希码，本文首先利用 DenseNet 网络提取图像的高维度特征，然后利用 PCA 技术对高维度特征进行降维，将同类图像降维后特征向量的平均值经过归一化后的结果作为该类图像的标签，其中标签的维度选择要求绝大部分标签之间的汉明距离 >ThL（阈值），目的是为了在检索过程中尽可能的减小检索到错误分类的可能性，ThL 的具体值根据实验需要设定，在本文的实验中，将 ThL 设置为 3。通过上述网络提取出来的 $H^{(1)}$ 会逼近预先设计好的目标哈

希码，考虑到方案的准确度，未对提取的 $H^{(1)}$ 进行归一化操作。

图像集中的每张图像 m_i 使用预训练的 CNN 模型提取的图像特征向量可表示为公式（3.1），其中 d_1 和 d_2 分别表示类别标签和图像特征维度，符号“||”表示连接。

$$H_i = (H_i^{(1)} || H_i^{(2)}) = (h_{i,1}, h_{i,2}, ..., h_{i,d_1}, f_{i,1}, f_{i,2}, ..., f_{i,d_2}) \tag{3.1}$$

3.3 基于随机矩阵的图像特征加密算法

为保证检索方案的安全性，通常需要对提取的特征向量进行加密，而基于同态加密的方案存在计算和存储成本大的问题，在实际应用中存在一定的局限性。本章提出了基于随机矩阵的图像特征加密算法，该算法支持密文域特征向量余弦相似度计算与排序，下面将对该算法的具体实现和安全性分析进行描述。

3.3.1 算法描述

假设图像集中的特征为 $f = (f_1, f_2, ..., f_{d_2})$，查询向量为 $q = (q_1, q_2, .., q_{d_2})$，其中 d_2 为图像特征的维度。首先令特征向量 $f' = (f'_1, f'_2, ..., f'_{d_2}) = \frac{1}{\|f\|}(f_1, f_2, ..., f_{d_2})$，随后将 f' 转换为 $d \times d$ 的矩阵 f''，其中 $d \times d = d_2$，具体转换方式如公式（3.2）所示。

$$f'' = \begin{bmatrix} f''_{1,1} & f''_{1,2} & \cdots & f''_{1,d} \\ f''_{2,1} & f''_{2,2} & \cdots & f''_{2,d} \\ \cdots & \cdots & \cdots & \cdots \\ f''_{d,1} & f''_{d,2} & \cdots & f''_{d,d} \end{bmatrix} = \begin{bmatrix} f'_1 & f'_2 & \cdots & f'_d \\ f'_{d+1} & f'_{d+2} & \cdots & f'_{2d} \\ \cdots & \cdots & \cdots & \cdots \\ f'_{(d-1)d+1} & f'_{(d-1)d+2} & \cdots & f'_{d_2} \end{bmatrix} \tag{3.2}$$

如公式（3.3）所示，接着图像所有者利用 $d \times d$ 维随机矩阵 U 将 f'' 进行扩展得到 $d \times (2d)$ 维矩阵 $\hat{f}$。

$$\hat{f} = (f''_{i,1}, U_{i,1}, f''_{i,2}, U_{i,2}, ..., f''_{i,d}, U_{i,d}), i \in [1, d] \tag{3.3}$$

最后使用 $(2d) \times (2d)$ 维随机可逆矩阵 M 和 $d \times (2d)$ 维随机矩阵 A 加密 $\hat{f}$ 获得密文 $\widetilde{f}$，具体加密方式如公式（3.4）所示。

$$\widetilde{f} = \hat{f} \times M^{-1} + A \tag{3.4}$$

检索用户在进行检索时，首先将检索向量 q 转换为 $q' = \frac{1}{\|q\|}(q_1, q_2, ..., q_{d_2})$，随后将 q' 按公式（3.2）的方法转换为 $d \times d$ 的矩阵 q''，并利用 $d \times d$ 维随机矩阵 V

对其进行扩展得到 $\hat{q}$，如公式（3.5）所示。

$$\hat{q} = (q''_{i,1}, V_{i,1}, q''_{i,2}, V_{i,2}, ..., q''_{i,d}, V_{i,d}), i \in [1, d] \tag{3.5}$$

最后生成查询陷门 $\widetilde{q}$ 并提交至云服务器进行相似度计算，加密方式如公式（3.6）所示。

$$\widetilde{q} = \hat{q} \times M^T + B^T \tag{3.6}$$

其中公式（3.6）中 B 为 $(2d) \times d$ 维矩阵，B 的第 i 行定义如公式（3.7）所示。

$$B_i = (M_{i,2}, M_{i,4}, ..., M_{i,2d}), i \in [1, 2d] \tag{3.7}$$

云服务器通过计算 $tr(\widetilde{f} \times \widetilde{q}^T)$ 的值对检索结果进行相似度排序，计算方式如公式（3.8）所示。

$$\begin{aligned} tr(\widetilde{f} \times \widetilde{q}^T) &= tr((\hat{f} \times M^{-1} + A) \times (\hat{q} \times M^T + B^T)^T) \\ &= tr(\hat{f} \times \hat{q}^T + \hat{f} \times M^{-1} \times B + A \times M \times \hat{q}^T + A \times B) \end{aligned} \tag{3.8}$$

明文状态下特征向量 f 和查询向量 q 的余弦相似度 $C(f, q)$ 可通过公式（3.9）计算得到。

$$C(f, q) = \sum_{i=1}^{d_2} \frac{f_i \times q_i}{\|f\| \|q\|} \tag{3.9}$$

公式（3.10）将公式（3.8）中的 $\hat{f} \times \hat{q}^T$ 进行展开计算可得。

$$\hat{f} \times \hat{q}^T = \begin{bmatrix} \frac{f_1}{\|f\|} & u_{1,1} & \cdots & \frac{f_d}{\|f\|} & u_{1,d} \\ \frac{f_{d+1}}{\|f\|} & u_{2,1} & \cdots & \frac{f_{2d}}{\|f\|} & u_{2,d} \\ \cdots & \cdots & \cdots & \cdots & \\ \frac{f_{(d-1)d+1}}{\|f\|} & u_{d,1} & \cdots & \frac{f_{d_2}}{\|f\|} & u_{d,d} \end{bmatrix} \times \begin{bmatrix} \frac{q_1}{\|q\|} & \frac{q_{d+1}}{\|q\|} & \cdots & \frac{q_{(d-1)d+1}}{\|q\|} \\ v_{1,1} & v_{2,1} & \cdots & v_{d,1} \\ \cdots & \cdots & \cdots & \cdots \\ v_{1,d} & v_{2,d} & \cdots & v_{d,d} \end{bmatrix} \tag{3.10}$$

根据计算和简化，可将 $tr(\hat{f} \times \hat{q})$ 的计算结果表示为公式（3.11）的形式。

$$tr(\hat{f} \times \hat{q}^T) = \sum_{i=1}^{d_2} \frac{f_i \times q_i}{\|f\| \|q\|} + \sum_{i=1}^{d} \sum_{j=1}^{d} u_{i,j} v_{i,j} \tag{3.11}$$

结合公式（3.8），公式（3.9）和公式（3.11）可知，基于随机矩阵的加密算法可保持密文特征向量和查询陷门余弦相似度顺序与明文余弦相似度顺序一致，从

而实现密文域余弦相似度的计算。

为了更好的理解上述算法，下面给出一个例子。假设图像集 $\mathcal{M}$ 中包含两张图像 m_1 和 m_2，检索用户需要检索与图像 m_q 最相似的一张图像。图像 m_1，m_2 和 m_q 的特征向量分别为 $f_1=(8,7,6,8)$，$f_2=(7,4,6,6)$，$q=(9,8,5,10)$。

步骤一：图像所有者生成 4×4 维的随机可逆矩阵 M，2×4 维随机矩阵 A 和 2×2 维随机矩阵 U，并计算得到密钥 B：

$$M=\begin{bmatrix}3&1&5&8\\4&7&7&9\\2&1&5&4\\4&7&8&2\end{bmatrix}\quad B=\begin{bmatrix}1&8\\7&9\\1&4\\7&2\end{bmatrix}\quad A=\begin{bmatrix}8&2&8&2\\2&8&5&7\end{bmatrix}\quad U=\begin{bmatrix}9&4\\5&2\end{bmatrix}$$

之后计算得到 M 的逆矩阵 M^{-1}：

$$M^{-1}=\begin{bmatrix}1.2286&-0.6171&-1.3886&0.64\\-0.2286&0.2171&-0.0114&-0.04\\-0.4&0.08&0.68&-0.12\\-0.0571&0.1543&0.0971&-0.16\end{bmatrix}$$

图像所有者首先计算得到 f_1' 和 f_2' 分别为：$f_1'=(0.5482,0.4796,0.4111,0.5482)$，$f_2'=(0.5981,0.3417,0.5126,5126)$，之后将 f_1' 和 f_2' 转换为 2×2 维的矩阵 f_1'' 和 f_2''：

$$f_1''=\begin{bmatrix}0.5482&0.4796\\0.4111&0.5482\end{bmatrix}\quad f_2''=\begin{bmatrix}0.5981&0.3417\\0.5126&0.5482\end{bmatrix}$$

接着使用随机矩阵 U 将 f_1'' 和 f_2'' 扩展为 $\hat{f}_1$ 和 $\hat{f}_2$。

$$\hat{f}_1=\begin{bmatrix}0.5482&9&0.4796&4\\0.4111&5&0.5482&2\end{bmatrix}\quad \hat{f}_2=\begin{bmatrix}0.5981&9&0.3417&4\\0.5126&5&0.5482&2\end{bmatrix}$$

最后通过公式（3.4）的方法加密获得密文矩阵:

$$\widetilde{f}_1=\begin{bmatrix}6.1959&4.2715&7.8507&1.2933\\1.0287&9.1844&4.9390&6.6773\end{bmatrix}\quad \widetilde{f}_2=\begin{bmatrix}6.3123&4.2297&7.6877&1.3417\\1.1676&9.1189&4.7739&6.7466\end{bmatrix}$$

步骤二：检索用户随机选择一个矩阵 V：

$$V=\begin{bmatrix}5&7\\3&9\end{bmatrix}$$

并按照公式（3.5）和公式（3.6）的方式求得查询陷门：

$$\widetilde{q} = \begin{bmatrix} 66.0775 & 110.5989 & 37.5298 & 62.0858 \\ 86.9558 & 116.4772 & 46.6515 & 47.0858 \end{bmatrix}$$

步骤三：云服务器计算得到 $tr(\widetilde{f_1} \times \widetilde{q}^T) = 2960.8074$，$tr(\widetilde{f_2} \times \widetilde{q}^T) = 2960.7769$，因为 $tr(\widetilde{f_2} \times \widetilde{q}^T) < tr(\widetilde{f_1} \times \widetilde{q}^T)$，所以云服务器认为 m_1 和 m_q 更加相似。而明文情况下，f_1 与 q 的余弦相似度为 $Sim(f_1, q) = 0.9924$，f_2 和 q 的余弦相似度为 $Sim(f_2, q) = 0.9618$，满足 $Sim(f_2, q) < Sim(f_1, q)$，与密文结果一致。

3.3.2 安全性分析

本小节中分析了基于随机矩阵的加密算法在已知密文攻击模型和已知背景攻击模型下的安全性。

推论 3.1 本算法可以在已知密文攻击模型下保证 f 和 q 的安全性。

证明 对特征向量 f 加密时，用于扩展的随机矩阵 U 以及密钥 M 和 A 由图像所有者保管，通过随机矩阵的保护，云服务器无法从加密后的 f 中获得明文信息。同时，检索用户生成查询陷门时使用的密钥 M 和 B 仅被授权的合法用户可知，且检索用户加密用的随机矩阵 V 在每次生成查询陷门时随机选择，对于云服务器也是未知的，云服务器也无法从查询陷门中获得查询的明文信息，因此本方案可以在已知密文模型下保护 f 和 q 的隐私安全。 □

推论 3.2 本方案可以在已知背景攻击模型下保证 f 和 q 的安全性。

证明 在已知背景模型中，敌手已知查询陷门和密文特征，其可能会通过线性攻击获得额外的背景信息，如尝试根据检索结果恢复明文信息。假设有查询向量 q 和两个图像特征 f_a 和 f_b，敌手可以构建等式（3.12）。

$$\begin{aligned} C_{a,b} &= tr(\widetilde{f_a} \times \widetilde{q}^T - \widetilde{f_b} \times \widetilde{q}^T) \\ &= tr[(\hat{f_a} \times \hat{q}^T + \hat{f_a} \times M^{-1} \times B) - (\hat{f_b} \times \hat{q}^T + \hat{f_b} \times M^{-1} \times B)] \\ &= tr[(\hat{f_a} - \hat{f_b}) \times (\hat{q}^T + M^{-1} \times B)] \end{aligned} \tag{3.12}$$

在公式（3.12）中，$\hat{f_a}$ 和 $\hat{f_b}$ 是原始 $d \times d$ 维特征矩阵 f_a 和 f_b 通过密钥 U 扩展得到的 $d \times 2d$ 的特征矩阵，其中包含 $2d_2$ 个未知数（$d \times d = d_2$）。在线性攻击中，敌手需要构建 $2d_2$ 个等式来求解 $2d_2$ 个未知数，而公式（3.12）中，用于扩展查询特征矩阵 $\hat{q}$ 的随机矩阵 V 是由检索用户每次检索时随机选择的，且矩阵 V 的维度为 $d \times d$，所以未知数增加至 $3d_2$，而通过 $2d_2$ 个等式求解 $3d_2$ 个未知数是不可行的。即使敌手获得了图像的明文和密文对，由于方案中的特征提取模型中的相

关参数是不公开的，敌手也无法根据明文图像得到对应的特征向量。此外，因为检索用户每次生成查询陷门时的随机矩阵 V 都是随机选择的，使得同一图像生成的查询陷门也不相同，保证了查询的不可链接性。因此，本方案可以在已知背景的模型下保证 f 和 q 的安全性。 □

3.4 基于 K-means 聚类算法的层次索引树构建方法

本方案选择使用基于 K-means 聚类的层次索引树作为检索结构。如图3.3所示，图像 m_i 和图像特征 $H_i^{(2)}$ 是一一对应的关系，使用 $H^{(1)}$ 部分进行 K-means 聚类得到 K 个聚类中心，并将聚类中心作为建树的叶节点。在第2.5章已经介绍过 K-means 聚类算法，在聚类时，初始值的随机选择对聚类结果存在一定的影响，为了使构建的检索索引结构稳定，本方案使用3.2.5小节中设计的目标哈希码作为 K-means 聚类的初始值并运行 K-means 聚类算法。

图像编号	图像特征向量
$ID(m_1)$	$H_1^{(2)}$
$ID(m_2)$	$H_2^{(2)}$
…	…
$ID(m_n)$	$H_n^{(2)}$

叶节点	图像编号
$leafNode_1$	$ID(m_5), ID(m_{19}), \dots, ID(m_{42}), ID(m_{119})$
$leafNode_2$	$ID(m_{13}), ID(m_{45}), \dots, ID(m_{76}), ID(m_{254})$
…	…
$leafNode_{ck}$	$ID(m_{28}), ID(m_{35}), \dots, ID(m_{510}), ID(m_{536})$

图 **3.3** 层次聚类的叶节点

本方案中构建索引树的过程如图3.4所示，索引树的每个节点都可以用一个三元组 $C_{h,i} = \{RV_{h,i}, E_{h,i}, IND_{h,i}\}$ 来表示，其中 h 代表第 h 层，i 是节点在第 h 层的索引，$RV_{h,i}$ 表示当前节点的代表向量，$E_{h,i}$ 包含了当前节点所有的孩子节点信息，$IND_{h,i}$ 是 $E_{h,i}$ 中节点间欧式距离的最大值，对于 K 个叶节点 $C_{0,1}, C_{0,2}, ..., C_{0,K}$，$C_{0,i} = \{ck_i, E_{0,i}, 0\}$。图像所有者每次选择欧式距离最近的两个节点合并，迭代，直到最终生成根节点。在该算法中，DM 表示两个 RV 的距离和两个节点的图像之间的最大成对距离的差异。$D(*,*)$ 表示两点之间的欧式距离。下面给出一个例子来帮助我们更好的理解层次聚类索引树的构建过程。

如图3.4所示，假设利用 $H^{(1)}$ 聚类得到 6 个聚类中心 $CK = \{ck_1, ck_2, ..., ck_6\}$。首先，我们选择距离最近的两个节点 $C_{0,1}$ 和 $C_{0,2}$，将它们合并成新的节点 $C_{1,1}$，节点 $C_{1,1}$ 的代表向量 $RV_{1,1} = \frac{RV_{0,1}+RV_{0,2}}{2}$，其 $IND_{1,1} = D(C_{0,1}, C_{0,2})$。同样的方法合并节点 $C_{0,4}$ 和 $C_{0,5}$ 生成节点 $C_{1,2}$。之后，最相似的两个节点是 $C_{1,1}$ 和 $C_{0,3}$，计算其 $DM = \frac{|D(C_{1,1},C_{0,3}) - max(IND_{1,1}, IND_{0,3})|}{max(IND_{1,1}, IND_{0,3})} = \frac{9-3}{3} = 2 > 0.5(Tht)$，所以我们合并 $C_{1,1}$ 和 $C_{0,3}$ 生成新的节点 $C_{2,1}$。接着，我们选择节点 $C_{1,2}$ 和 $C_{0,6}$，计算 $DM = \frac{D(C_{1,2},C_{0,6}) - max(IND_{1,2}, IND_{0,6})}{max(IND_{1,2}, IND_{0,6})} = \frac{13-10}{10} = 0.3 < 0.5(Tht)$，所以选择合并节点 $C_{0,6}$ 到节点 $C_{1,2}$ 并更新节点 $C_{1,2}$ 的信息。迭代上述算法直到层次索引树构建完

成，算法中的阈值 Tht 根据实验结果设置。

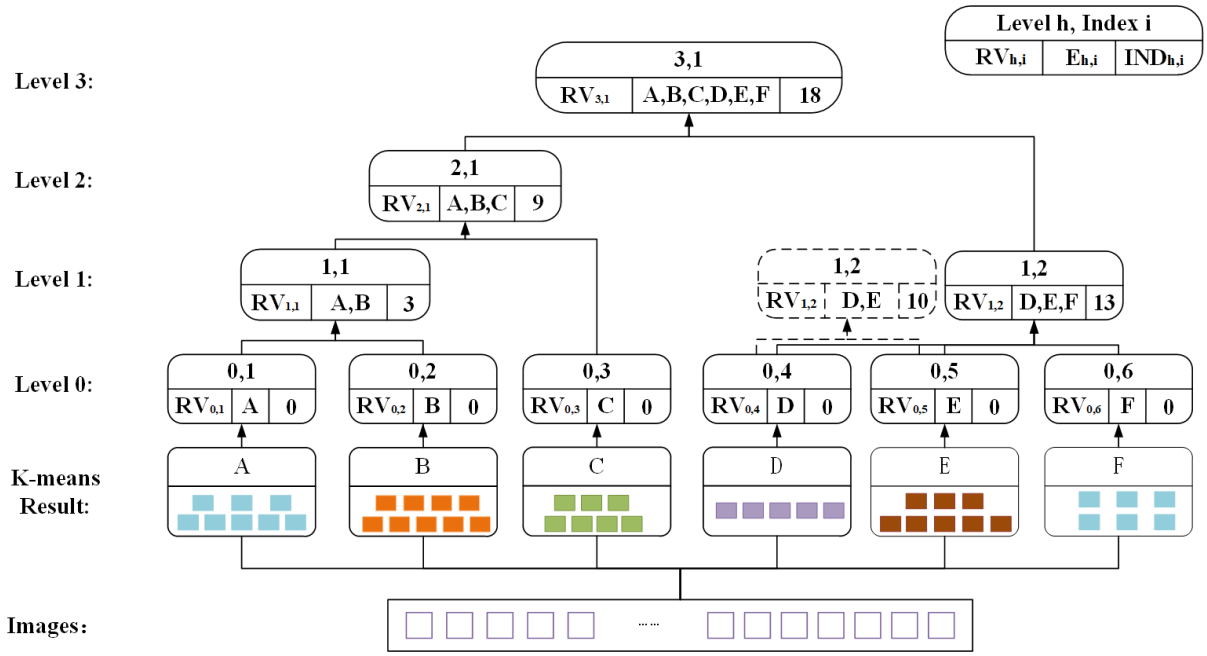


图 **3.4**　层次聚类索引树结构

算法1给出了生成层次索引树叶节点的伪代码，该算法通过特征向量集合 $\mathcal{H}$ 生成叶节点集合 C_{set}。

算法 1 生成层次索引树叶节点

输入： 图像特征向量集合 $\mathcal{H}$，聚类中心数量 K

输出： 叶节点集合 C_{set}

$C_{set} \leftarrow \emptyset$
$CK \leftarrow Kmeans(H_i^{(1)}, K)$ /*$CK = \{ck_1, ck_2, ..., ck_K\}$*/
for $i = 1$ *to* K **do**
    $p \leftarrow Node()$
    $p.RV \leftarrow ck_i$
    for $j = 1$ *to* n **do**
        if $H_i^{(1)}$ *in* ck_i **then**
            $p.child.append(H_i^{(2)})$
        end if
    end for
    $p.IND = 0$
    $C_{set}.append(p)$
end for
return C_{set}

算法 2 给出了构建索引树的伪代码，该算法以叶节点集合 C_{set} 和阈值 Tht 作为输入，并输出生成的层次索引树 $\mathcal{I}$。

算法 **2** 构建层次索引树

输入**:** 叶节点集合 C_{set}，阈值 $Tht.$

输出**:** 层次索引树 $\mathcal{I}$

$\mathcal{I} \leftarrow \emptyset$
$Insert(\mathcal{I}, C_{set})$ //把生成的节点插入索引树中
while $len(C_{set}) > 1$ **do**
  $C_a, C_b \leftarrow find\_nearest(C_{set})$ /* 返回 C_{set} 中距离最小的两个节点 $C_a = C_{h1,i1}$ 和 $C_b = C_{h2,i2}$，其中 C_a 和 C_b 满足 $(h1 > h2)$ 或者 $(h1 == h2$ 且 $i1 > i2)$*/
  if $C_a.IND == 0\ and\ C_b.IND == 0$ **then**
    $C_{set} \leftarrow Merge(\mathcal{I}, C_{set}, C_a, C_b, True)$
  else
    $DM \leftarrow \frac{D(C_a,C_b) - max(C_a.IND,\ C_b.IND)}{max(C_a.IND,\ C_b.IND)}$
    if $DM > Tht$ **then**
      $C_{set} \leftarrow Merge(\mathcal{I}, C_{set}, C_a, C_b, True)$
    else
      $C_{set} \leftarrow Merge(\mathcal{I}, C_{set}, C_a, C_b, False)$
    end if
  end if
end while
return $\mathcal{I}$

function $Merge(\mathcal{I}, C_{set}, C_a, C_b, flag)$
    if $flag$ **then**
      $C_{new} = (RV_{new}, E_{new}, IND_{new}) = CreateNode(C_a, C_b)$ //生成新节点
      $Insert(\mathcal{I}, C_{new})$
      $C_{set}.append(C_{new})$
      remove C_a, C_b from C_{set} //将节点 C_a, C_b 从集合 C_{set} 中移出。
    else
      $C_a.append(C_b)$
      remove C_b from C_{set}
    end if
    return C_{set}

3.5 基于深度学习的图像可搜索加密方案

如图3.5所示，本方案共包括三个实体和六个算法部分，图像所有者主要完成密钥生成（GenKey）、加密图像（EncImage）、生成安全索引（GenIndex）三个部分，检索用户需要生成查询（GenQuery）和解密图像（DecImage），云服务器则需要承担存储和检索（Search）任务，下面将对各个实体的任务进行详细描述。

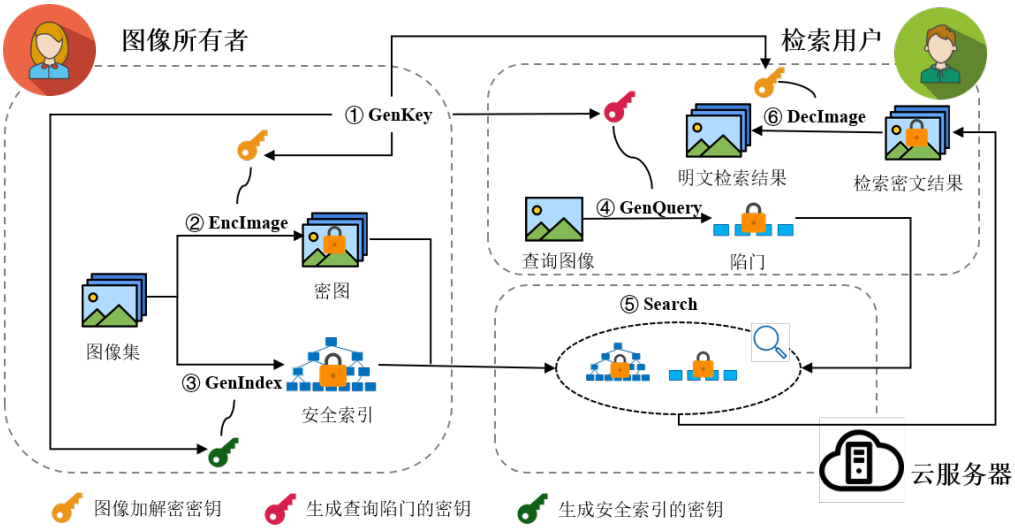


图 **3.5** 系统框架

3.5.1 图像所有者

（1）密钥生成（GenKey）

图像所有者生成的密钥主要应用于图像加解密以及索引和查询加密。图像所有者生成两个 $(2d_1)\times(2d_1)$ 维随机可逆矩阵 M_O 和 M_U 且满足 $M_O\times M_U=I$，其中 I 表示单位矩阵。A 和 U 分别是 $d\times(2d)$ 和 $d\times d$ 的随机矩阵，M 是 $(2d)\times(2d)$ 维的随机可逆矩阵。密钥 pk 是图像加密密钥，sk 为图像解密密钥。B 是 $(2d)\times d$ 维的矩阵，其中矩阵 B 的第 i 行定义如公式（3.13）所示。

$$B_i=(M_{i,2},M_{i,4},...,M_{i,2d}),i\in[1,2d] \tag{3.13}$$

图像所有者生成的密钥 $\psi_O=\{M_O,M^{-1},A,U,pk,\gamma\}$ 用于加密图像和生成安全索引，$\psi_U=\{M_U,M,B,sk,\gamma\}$ 发送给授权用户生成查询陷门和解密检索结果。

（2）图像加密（EncImage）

对于图像的加密，如公式（3.14）所示，图像集 $\mathcal{M}$ 中的 n 张图像使用密钥 pk 进行加密，得到加密后的图像集 $\mathcal{C}$。

$$\mathcal{C}=Enc(\mathcal{M},pk) \tag{3.14}$$

（3）生成安全索引（GenIndex）

索引的生成具体可划分为三部分：特征提取、构建明文索引和加密索引。

①特征提取：对于图像集 $\mathcal{M}$ 中的每张图像 m_i 都能通过预训练的 CNN 模型得到其特征向量 $H_i = (H_i^{(1)}||H_i^{(2)}) = (h_{i,1}, h_{i,2}, ..., h_{i,d_1}, f_{i,1}, f_{i,2}, ..., f_{i,d_2}), i \in [1, n]$。

②构建明文索引：本方案利用 $H^{(1)}$ 进行 K-means 聚类，并将得到的聚类中心作为叶节点构建了层次索引树，具体的建树算法在3.4节中已经详细介绍。

③加密索引：本文中的特征向量 H_i 由 $H_i^{(1)}$ 和 $H_i^{(2)}$ 两部分组成，对这两部分分别采用基于 LWE 的安全 kNN 算法和基于随机矩阵的加密算法。

对于所有 $H_i^{(1)} = (h_{i,1}, h_{i,2}, ..., h_{i,d_1})$，图像所有者首先按照公式（3.15）所示的方法将 $H_i^{(1)}$ 扩展为 $\widetilde{H_i^{(1)}}$。

$$\widetilde{H_i^{(1)}} = (h_{i,1}, h_{i,2}, ..., h_{i,d_1}, -\frac{1}{2}\sum_{j=1}^{d_1} h_{i,j}, \alpha_1, \alpha_2, ..., \alpha_{d_1-1}) \tag{3.15}$$

其中 $\alpha_1, \alpha_2, ..., \alpha_{d_1-1} \in \mathbb{Z}_{p_2}$ 由图像所有者随机选择。随机数 $\gamma \in \mathbb{Z}_{p_1}$，$p_1$ 和 p_2 满足 $p_1 \gg p_2$，且 $\gamma \gg 2|max(\vec{\xi_i})|$，$\vec{\xi_i} \in \mathbb{Z}_{p_2}^{2d_1}$ 为随机整数噪声向量。

然后图像所有者使用公式（3.16）的方式加密 $\widetilde{H_i^{(1)}}$ 获得密文 $Enc_{M_O}(\widetilde{H_i^{(1)}})$。

$$Enc_{M_O}(\widetilde{H_i^{(1)}}) = (\gamma \cdot \widetilde{H_i^{(1)}} + \vec{\xi_i}) \times M_O \tag{3.16}$$

当加密 $H_i^{(2)}$ 时，图像所有者首先根据公式（3.17）将 $H_i^{(2)}$ 转换为 $H_i^{(2)'}$。

$$H_i^{(2)'} = \frac{1}{\|H_i^{(2)}\|}(f_{i,1}, f_{i,2}, ..., f_{i,d_2}), i \in [1, n]. \tag{3.17}$$

如公式（3.18）所示，接着图像所有者将 $H_i^{(2)'}$ 转换为 $d \times d$ 维的矩阵 $\overline{H_i^{(2)}}$。

$$\overline{H_i^{(2)}} = (\overline{f_{i,j,1}}, \overline{f_{i,j,2}}, ..., \overline{f_{i,j,d}}), i \in [1, n], j \in [1, d] \tag{3.18}$$

然后将 $\overline{H_i^{(2)}}$ 按公式（3.19）扩展为 $d \times (2d)$ 维的矩阵 $\widetilde{H_i^{(2)}}$。

$$\widetilde{H_i^{(2)}} = (\overline{f_{i,j,1}}, U_{j,1}, \overline{f_{i,j,2}}, U_{j,2}, ..., \overline{f_{i,j,d}}, U_{j,d}), i \in [1, n], j \in [1, d] \tag{3.19}$$

最后，图像所有者使用随机可逆矩阵 M 和随机矩阵 A 加密 $\widetilde{H_i^{(2)}}$ 获得最终密文 $Enc(\widetilde{H_i^{(2)}})$，具体加密方式如公式（3.20）所示。

$$Enc(\widetilde{H_i^{(2)}}) = \widetilde{H_i^{(2)}} \times M^{-1} + A \tag{3.20}$$

完成 GenIndex 算法后，图像所有者将密文图像和安全索引上传到云服务器。

3.5.2 检索用户

（1）陷门生成（GenQuery）

检索用户进行检索时，首先利用预训练的 CNN 模型提取检索图像 m_q 的特征向量：$H_q = (H_q^{(1)}||H_q^{(2)}) = (h_{q,1}, h_{q,2}, ..., h_{q,d_1}, f_{q,1}, f_{q,2}, ..., f_{q,d_2})$。

在对 $H_q^{(1)}$ 加密时，检索用户首先将 $H_q^{(1)}$ 扩展为 $2d_1$ 维的向量 $\widetilde{H_q^{(1)}}$，如公式（3.21）所示，其中 $\beta_1, \beta_2, ..., \beta_{d_1-1} \in \mathbb{Z}_{p_2}$ 均为随机数，$\delta_q \in \mathbb{Z}_{p_2}$ 为随机正整数。

$$\widetilde{H_q^{(1)}} = (\delta_q h_{q,1}, \delta_q h_{q,2}, ..., \delta_q h_{q,d_1}, \delta_q, \beta, \beta_1, \beta_2, ..., \beta_{d_1-1}) \tag{3.21}$$

然后，检索用户使用随机可逆矩阵 M_U 和随机数 γ 加密扩展的向量，其中 $\vec{\xi_q} \in \mathbb{Z}_{p_2}^{2d_1}$ 是检索用户随机选择的整数扰动，具体加密方式如公式（3.22）所示。

$$Enc(\widetilde{H_q^{(1)}}) = M_U \times (\gamma \widetilde{H_q^{(1)}}^T + \vec{\xi_q}^T) \tag{3.22}$$

对于 $H_q^{(2)}$，如公式（3.23）所示，检索用户首先将 $H_q^{(2)}$ 转换为 $H_q^{(2)'}$。

$$H_q^{(2)'} = \frac{1}{\| H_q^{(2)} \|}(f_{q,1}, f_{q,2}, ..., f_{q,d_2}) \tag{3.23}$$

之后，将 $H_q^{(2)'}$ 转换为公式（3.24）中 $d \times d$ 维的矩阵 $\overline{H_q^{(2)}}$。

$$\overline{H_q^{(2)}} = (\overline{f_{q,j,1}}, \overline{f_{q,j,2}}, ..., \overline{f_{q,j,d}}), j \in [1, d] \tag{3.24}$$

然后按公式（3.25）中的方式将 $\overline{H_q^{(2)}}$ 扩展为 $d \times (2d)$ 维的矩阵，其中 V 为检索用户随机选择的 $d \times d$ 维的随机矩阵。

$$\widetilde{H_q^{(2)}} = (\overline{f_{q,j,1}}, V_{j,1}, \overline{f_{q,j,2}}, V_{j,2}, ..., \overline{f_{q,j,d}}, V_{j,d}), j \in [1, d] \tag{3.25}$$

最后，检索用户使用随机可逆矩阵 M 和矩阵 B 加密 $\widetilde{H_q^{(2)}}$ 获得公式（3.26）中的密文，并将生成的查询陷门上传到云服务器进行检索。

$$Enc(\widetilde{H_q^{(2)}}) = \widetilde{H_q^{(1)}} \times M^T + B^T \tag{3.26}$$

（2）图像解密（DecImage）

如公式（3.27）所示，检索用户在收到来自云服务器返回的检索结果 $\mathcal{R}'$ 时，利用密钥 sk 解密获得明文图像集 $\mathcal{R}$。

$$\mathcal{R} = DecImage(\mathcal{R}', sk) \tag{3.27}$$

3.5.3 云服务器

除了存储安全索引和密文图像，云服务器还需要进行相似度计算，即从密文图像集中检索出与查询图像相似度最高的 k 张图像返回给检索用户。本方案的检索过程可细分为两部分：（1）检索到正确分类；（2）检索到相似图像。

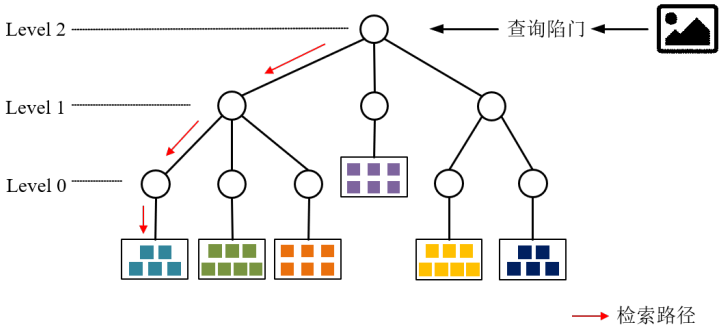


图 **3.6** 检索示意图

Step 1：如图3.6所示，在接收到来自检索用户的检索请求时，云服务器首先使用加密的 $H_q^{(1)}$ 的和安全索引 $\widetilde{I}$ 检索到正确分类。每次检索均从根节点出发，通过计算节点与加密的 $H_q^{(1)}$ 的欧式距离选择进入到的下一层的节点，直到最终找到叶节点，即找到了最相似的分类中，具体计算方式如公式（3.28）所示。

$$\begin{aligned} Sim_1(H_i^{(1)}, H_q^{(1)}) &= \left\lceil \frac{Enc(H_i^{(1)}) \times Enc(H_q^{(1)})}{\gamma^2} \right\rfloor_{p_1} \\ &= \left\lceil \frac{(\gamma \widetilde{H_i^{(1)}} + \vec{\xi_i}) \times M_O \times M_U \times (\gamma \widetilde{H_q^{(1)}}^T + \vec{\xi_q}^T)}{\gamma^2} \right\rfloor_{p_1} \\ &= \sum_{k=1}^{d_1} \delta_q \cdot h_{i,k} \cdot h_{q,k} - \frac{\delta_q}{2} \sum_{k=1}^{d_1} h_{i,k}^2 + \sum_{k=1}^{d_1} \alpha_k \beta_k \\ &= -\frac{\delta_q}{2} (\| H_i^{(1)} - H_q^{(1)} \|^2 - \| H_q^{(1)} \|^2) + \sum_{k=1}^{d_1} \alpha_k \beta_k \end{aligned} \tag{3.28}$$

Step 2：通过 Step 1 找到正确分类后，云服务器通过公式（3.29）计算相似度进行相似度排序，并返回相似度最高的 k 张图像。

$$Sim_2(H_i^{(2)}, H_q^{(2)}) = tr(Enc(\widetilde{H_i^{(2)}}) \times Enc(\widetilde{H_q^{(2)}})^T) \tag{3.29}$$

3.6 安全性分析

本章节将对方案的安全性进行详细分析，主要从图像安全性、索引和查询的安全性以及查询的不可链接性三个方面进行分析论证，其中，图像特征采用基于随机矩阵的加密算法，该算法的安全性已经在3.3.2节被证明。

3.6.1 图像的安全性

在本方案中，只有拥有图像解密密钥 sk 的合法用户才能解密密文图像。因此，本方案中图像的安全性能够得到保证。

3.6.2 类别标签的安全性

本方案采用基于 LWE 的安全 kNN 算法对 $H^{(1)}$ 进行加密，该加密算法由 Yuan 等人[19] 基于误差学习问题提出，其安全性已经被证明，下面我们将回顾该方案的安全性证明。首先给出 LWE 问题的定义，然后证明 $H_i^{(1)}$ 和 $H_q^{(1)}$ 在已知密文攻击模型和已知背景攻击模型下的安全性。

定义 **3.1** LWE 问题[18]：对于多项式（$\vec{v_i} \in \mathbb{Z}_{p_1}^{d_1}$，$w_i \in \mathbb{Z}_{p_1}$）的多个样本

$$w_i = \vec{f} \times v_i^T + \xi_i \tag{3.30}$$

且误差 $\xi_i \in \mathbb{Z}_{p_1}$ 从某个概率分布中得到，如果想要以不可忽略的概率恢复 $\vec{f}$ 在计算上不可行。

推论 3.3 本方案可以在已知密文攻击模型下保护 $H_i^{(1)}$ 和 $H_q^{(1)}$ 的安全性

证明 在本方案中，每个扩展向量 $\widetilde{H_i^{(1)}}$ 通过如下方式加密：

$$\begin{aligned} Enc_{M_O}(\widetilde{H_i^{(1)}}) &= (\gamma \cdot \widetilde{H_i^{(1)}} + \vec{\xi_i}) \times M_O \\ &= \gamma \cdot \widetilde{H_i^{(1)}} \times M_O + \vec{\xi_i} \times M_O \end{aligned} \tag{3.31}$$

如公式（3.31）所示，$\widetilde{H_i^{(1)}}$ 和 $\vec{\xi_i}$ 均为 $2d_1$ 维的向量，其与矩阵 M_O 的乘积可以被看做 $4d_1^2$ 次 $2d_1$ 维的向量点积。用 $Enc_{M_O}(\widetilde{H_i^{(1)}})(j)$ 表示 $Enc_{M_O}(\widetilde{H_i^{(1)}})$ 的第 j 个元素，$M_O(j)$ 表示矩阵 M_O 的第 j 列，且 $1 \leqslant j \leqslant 2d_1$。我们将 $\gamma \cdot M_O(j)$ 定义为 $M_O(j)'$，$\vec{\xi_i} \times M_O(j)$ 表示为 $\xi_{i,j}'$，可得到公式（3.32）中 $2d_1$ 个关于 $(M_O(j)', Enc_{M_O}(\widetilde{H_i^{(1)}})(j))$ 的样本。

$$Enc_{M_O}(\widetilde{H_i^{(1)}})(j) = \widetilde{H_i^{(1)}} \times M_O(j)' + \xi_{i,j}', 1 \leqslant j \leqslant 2d_1 \tag{3.32}$$

已知密文 $Enc_{M_O}(\widetilde{H_i^{(1)}})$ 的情况下恢复 $\widetilde{H_i^{(1)}}$ 转换成了我们上面描述的 LWE 问题，而求解 LWE 问题在计算上是不可行的，并且本方案中的密钥 M_O 对于敌手是保密的，M_O' 对于敌手也是未知的。因此，密钥 M_O 和 $\vec{\xi_i}$ 保密的情况下，敌手在已知密文攻击模型下从 $Enc_{M_O}(\widetilde{H_i^{(1)}})$ 恢复 $\widetilde{H_i^{(1)}}$ 比求解 LWE 问题困难。同样，$H_q^{(1)}$ 的加密方式也可以转换为上述的 LWE 问题。因此，本方案可以保护已知密文攻击模型下 $H_i^{(1)}$ 和 $H_q^{(1)}$ 的安全。 □

推论 3.4 本方案可以在已知背景攻击模型下保护 $H_i^{(1)}$ 和 $H_q^{(1)}$ 的安全性。

证明 上面已经证明敌手在已知索引和查询向量密文的情况下无法恢复出明文。在已知背景模型中，敌手可以获得查询陷门和密文特征，他可以通过线性攻击获得额外的背景信息，如尝试从检索结果中恢复明文信息。假设有查询向量 $H_q^{(1)}$ 和两个索引树的节点向量 $H_a^{(1)}$ 和 $H_b^{(1)}$，敌手可以构建等式（3.33）。

$$D_{a,b} = \frac{\delta_q}{2}(\|H_a^{(1)} - H_q^{(1)}\|^2 - \|H_b^{(1)} - H_q^{(1)}\|^2) \tag{3.33}$$

假设敌手已知查询 $H_q^{(1)}$，那么在公式3.33中则存在 $2d_1+1$ 个未知量，线性分析攻击要求敌手获得 $2d_1$ 个以上的查询向量并构建 $2d_1$ 个上述等式求解节点向量 $H_a^{(1)}$ 和 $H_b^{(1)}$。但是对于每个查询向量在加密的时候 δ_q 是不同的，对于 $2d_1$ 个等式中的 $4d_1$ 个未知数，敌手无法求解。因此，加密后的 $H_i^{(1)}$ 和 $H_q^{(1)}$ 可以抵御已知背景模型攻击。 □

3.6.3 查询的不可链接性

推论 3.5 本方案能够保证查询的不可链接性，即相同的图像，每次生成的查询陷门不相同。

证明 假设向量 $H_{q,1}$ 和 $H_{q,2}$ 是利用预训练的CNN模型从同一查询图像 m_q 中提取得到的查询向量。通过上述算法描述可知，用于加密 $H_{q,1}^{(1)}$ 和 $H_{q,2}^{(1)}$ 的随机数 δ 和扩展的随机向量 β 是随机的，同时在公式（3.34）中的向量 $\vec{\xi_{q,1}}$ 和 $\vec{\xi_{q,2}}$ 也不相同，通过上述操作加密得到的密文 $Enc_{M_U}(\widetilde{H_{q,1}^{(1)}})$ 和 $Enc_{M_U}(\widetilde{H_{q,1}^{(2)}})$ 将完全不同，因此，同一查询图像中提取的 $H_q^{(1)}$ 经过加密后的密文不相同。

$$\begin{aligned} Enc_{M_U}(\widetilde{H_{q,1}^{(1)}}) &= M_U \times (\gamma \cdot \widetilde{H_{q,1}^{(1)}}^T + \vec{\xi_{q,1}}^T) \\ Enc_{M_U}(\widetilde{H_{q,2}^{(1)}}) &= M_U \times (\gamma \cdot \widetilde{H_{q,2}^{(1)}}^T + \vec{\xi_{q,2}}^T) \end{aligned} \tag{3.34}$$

对于图像特征 $H_q^{(2)}$，如公式（3.35）所示，检索用户加密时用于扩展的矩阵 V 每次都是随机选择，因此扩展加密后的 $\widetilde{H_{q,1}^{(2)}}$ 和 $\widetilde{H_{q,2}^{(2)}}$ 不同，即：同一张图像 m_q 每次生成的密文 $H_q^{(2)}$ 不同。

$$\begin{aligned} \widetilde{H_{q,1}^{(2)}} &= (\overline{f_{q,j,1}}, V_{q_1,1}, \overline{f_{q,j,2}}, V_{q_1,2}, ..., \overline{f_{q,j,d}}, V_{q_1,d}) \\ \widetilde{H_{q,2}^{(2)}} &= (\overline{f_{q,j,1}}, V_{q_2,1}, \overline{f_{q,j,2}}, V_{q_2,2}, ..., \overline{f_{q,j,d}}, V_{q_2,d}) \end{aligned} \tag{3.35}$$

因此，即使是相同图像 m_q 生成的查询陷门也不同，敌手无法根据查询陷门判断查询图像的关联。综上所述，本文提出的方案能保证查询的不可链接性。 □

3.7 实验评估

本文通过在现有数据集上进行实验来对方案的性能进行分析和评估，进一步论证方案的可行性。本章节的实验均在一台装有 Windows10 操作系统、3.60GHZ Intel Corel i7CPU 和 8GB 内存的 PC 机上运行，使用的编程语言为 Python，并选用 Caltech256 图像集[64] 作为实验数据集，该图像集是加利福尼亚理工学院收集整理的数据集，共包括 256 类，每个类别图像超过 80 张。同时，本方案使用 Precision at top-k($P@k$) 作为本文检索准确度的衡量指标，$P@k$ 的定义如公式（3.36）所示。

$$P@k = \frac{correct\_num}{k} \tag{3.36}$$

其中 $correct\_num$ 表示返回的 top-k 张图像中正确的图像数量。

本文使用两段式的网络模型提取图像特征向量，在 Caltech256 中随机选择了 50 类、100 类、150 类和 200 类图像。在进行网络训练时，每类图像包含训练集 60 张，验证集 20 张。进行检索时，每类图像集包含 80 张图像，测试集包含 20 张图像。由于部分图像集中图像总数少于 100 张，可能会存在少量图像重复的情况。下面将从维度选择、检索效率、检索准确度、存储成本等方面与现有的三类方案 SEI[22]，SVMIS[54] 和 KNN-CBIR 进行对比来对本方案进行评估，其中对比方案 KNN-CBIR 采用2.3.3小节中所述的安全 KNN 算法加密特征向量。

3.7.1 维度选择

图像特征向量的维度对于检索效率，存储成本和检索准确度都存在影响。通常高维度的特征能实现较高的检索准确度，但这也会降低检索效率并增加存储成本。因此，选择的特征维度应在保证检索准确度的前提下尽可能的短。本文提取的特征向量包含 $H^{(1)}$ 和 $H^{(2)}$ 两部分。$H^{(1)}$ 的维度选择采用3.2.5节中的选择策略，在图像类别为 50 类、100 类、150 类和 200 类时，本方案分别选择 $H^{(1)}$ 维度为 24、32、48、64 维。对于 $H^{(2)}$ 的选择，在 50 类图像时，分别计算 $H^{(2)}$ 维度为 36、64、121、256 维时的检索准确度，如表3.3所示，综合维度和准确度的考虑，本方案最终选定 $H^{(2)}$ 的维度为 64 维。

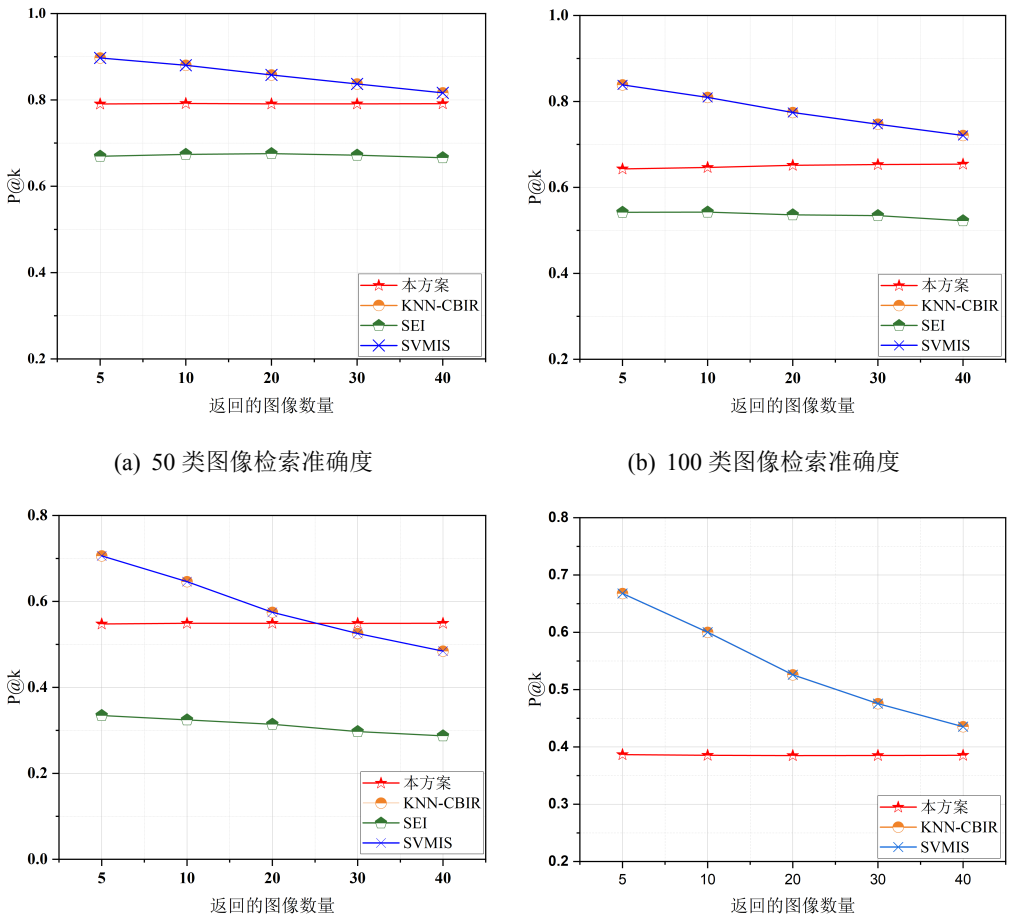
表 **3.3** **50** 类图像时不同维度 $H^{(2)}$ 的检索准确度

特征维度	top-k				
	5	10	20	30	40
36	0.9246	0.9235	0.9224	0.9224	0.9218
64	**0.9362**	**0.9343**	**0.9327**	**0.9314**	**0.9307**
121	0.9368	0.9334	0.9319	0.9302	0.9285
256	0.9458	0.9423	0.9392	0.9378	0.9367

四类方案均采用 CNN 模型提取图像特征，其中 SEI、SVMIS 和 KNN-CBIR 均提取 512 维特征向量后使用 PCA 技术降维至 32、64、128、256 和 512 维，实验表明在 128 维时检索准确度最优，因此选取 128 维的特征向量用于实验，且特征维度不随图像类别数量变化。而本方案图像特征向量维度与图像类别相关，$H^{(1)}$ 的维度随图像类别数量增加，在 50 类、100 类、150 类和 200 类图像时，提取的特征维度分别为 24+64、32+64、48+64 和 64+64。同时，在四类方案中，SVMIS 和 KNN-CBIR 为线性检索，本方案与 SEI 为树形检索结构。

3.7.2 检索准确度

在加密图像检索中，检索的准确度与特征提取方法、特征维度、加密算法有关，并且随着图像类别的增加，检索的准确度也会降低，因为图像类别数量越多，越容易检索到错误分类的图像。本实验取 $P@k$ 衡量指标中的 k 分别为 5、10、20、30、40 进行了比较。



(a) 50 类图像检索准确度　(b) 100 类图像检索准确度

(c) 150 类图像检索准确度　(d) 200 类图像检索准确度

图 **3.7**　多类别图像检索准确度对比

如图3.7所示，我们比较了图像类别为 50 类、100 类、150 类和 200 类时，本方案与 SEI、SVMIS 以及 KNN-CBIR 三类方案的检索准确度。通过对比可以看到，

线性检索方案通过逐一匹配，能实现比树形检索结构更高的检索准确度。在 50 类图像时，SVMIS 在返回 top-40 时的准确度高达 81%，明显高于本方案。对于本方案，树形索引构建过程中参数 Tht 的选取对于索引树的结构有很大的影响，Tht 选择不当可能造成索引树结构不平衡从而影响检索准确度，因此，选择合适的 Tht 很重要。本方案和 SEI 均属于树形检索结构，本方案使用 $H^{(1)}$ 进行 K-means 聚类，而 $H^{(1)}$ 通过特征提取网络模型的学习能够使得聚类更加准确。同时，我们选用余弦相似度作为 $H^{(2)}$ 的相似性衡量指标，整体方案能够实现比 SEI 更高的检索准确度。在图像类别数量为 200 时，本方案返回 top-40 的准确度约为 38.6%，略低于线性检索方案 43.5% 的检索准确度。

3.7.3 时间成本和存储代价

1. 时间成本

本小节对方案中的安全索引构造时间和检索时间进行了对比分析。

（1）安全索引构建：构建安全索引的时间开销与索引结构以及特征和索引的加密算法有关。本方案计算的安全索引构造时间包含构造明文索引和加密索引的时间。SEI 使用 AP 聚类和 K-means 聚类相结合的 CAK-means 聚类算法，使得聚类更准确的同时也产生了较长的索引构建时间。方案 SVMIS 和 KNN-CBIR 是线性索引结构，安全索引的构造时间即加密特征向量的时间，因此安全索引构造的时间开销最小。本方案采用 $H^{(1)}$ 构建基于 K-means 聚类的层次索引树，并采用基于 LWE 的安全 kNN 算法和基于随机矩阵的加密算法分别对 $H^{(1)}$ 和 $H^{(2)}$ 加密。如图3.8所示，我们比较了 50 类、100 类、150 类和 200 类图像时各类方案构造安全索引所花的时间，其中方案 SEI 在 200 类图像时因为聚类时间和所占内存开销较大未进行比较。

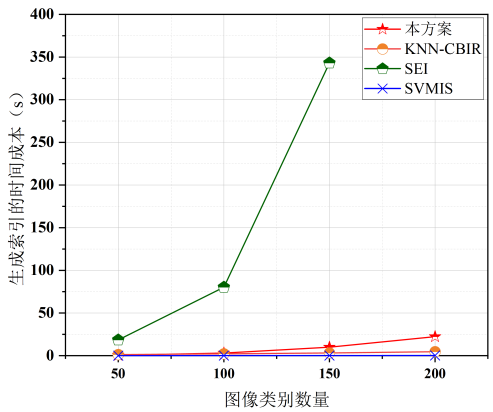
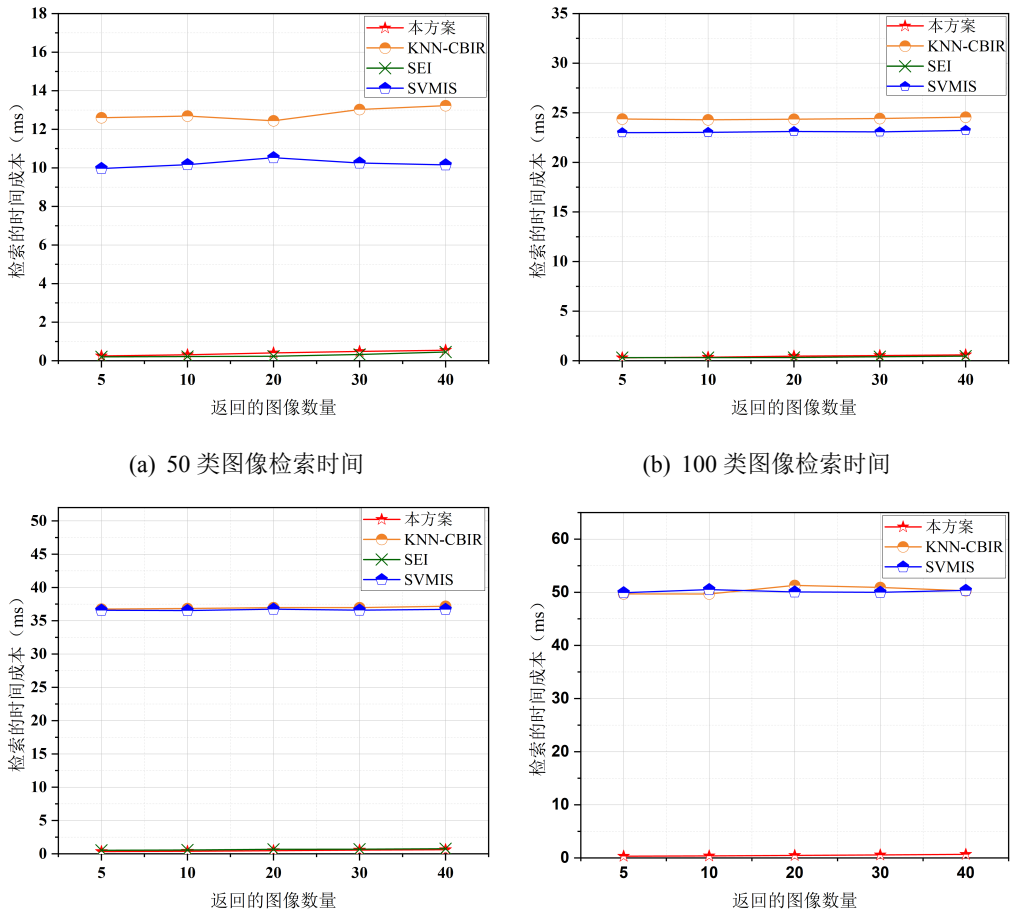


图 **3.8**　多类别图像生成索引的时间开销对比

从图3.8我们可以看出，SVMIS 生成安全索引的时间最短，而本方案构建安全索引的时间开销远远小于树形检索方案 SEI。因为本方案的聚类算法中使用的 K-means 聚类算法时间复杂度小于 CAK 聚类算法，同时本方案采用 $H^{(1)}$ 进行聚

类，在 200 类图像时，$H^{(1)}$ 的维度为 64 维，低于 SEI 中使用的 128 维。

（2）检索：云服务器收到检索用户的检索请求后，首先利用查询陷门的 $H^{(1)}$ 检索到具体分类中，此过程采用公式（3.28）中的计算方式。之后在分类中利用 $H^{(2)}$ 进行细粒度的逐一匹配，计算方式如公式（3.29）所示。我们比较了四类方案的检索效率，如图3.9所示。从图中我们可以看出，线性检索方案 SVMIS 和 KNN-CBIR 通过与图像集中的所有特征向量逐一匹配进行检索，检索效率明显低于两类树形检索方案，但是随着图像数量的增加，方案 SVMIS 和 KNN-CBIR 的检索效率明显下降。而树形检索的方案 SEI 和本文提出的方案在多类别图像时均能保持较高的检索效率，在图像类别数量为 50 类时，本方案返回 top-40 约花费 0.5468ms，在图像类别为 200 类时，返回 top-40 约花费 0.6717ms。



(a) 50 类图像检索时间　　(b) 100 类图像检索时间

(c) 150 类图像检索时间　　(d) 200 类图像检索时间

图 **3.9**　多类别图像检索效率对比

2. 存储代价

本小节中考虑的存储成本主要包含密钥、安全索引、密图和预训练的 CNN 模型所占的存储空间。其中密钥和预训练的 CNN 模型分别由图像所有者和授权用户各自保存，安全索引和密图由云服务器存储。对比的三类方案特征维度为 128 维时检索准确度最优，SEI 和 SVMIS 通过加密后分别为 258 维和 256 维，而

本方案在 200 类图像时，采用 64+64 维的特征，加密后的 $H^{(1)}$ 为 128 维，加密后的 $H^{(2)}$ 为 8×16 维的矩阵。

本文比较了各个方案的预训练 CNN 模型的存储成本，本方案使用的两段式网络模型在 50 类图像时约占 98.7745MB 存储空间，模型存储空间随图像类别变化很小。SEI、SVMIS 和 KNN-CBIR 均提取 512 维特征向量后进行 PCA 降维，模型存储成本约为 188.7849MB。此外，我们还计算了本方案密钥的存储成本，在 200 类图像时，密钥存储成本约为 0.2558MB。

如图3.10所示，我们对比了本方案与 SEI、SVMIS 和 KNN-CBIR 三类方案在 50、100、150 和 200 类图像时安全索引的存储成本，从图中我们可以看出，本方案是四个方案中安全索引存储成本最低的。

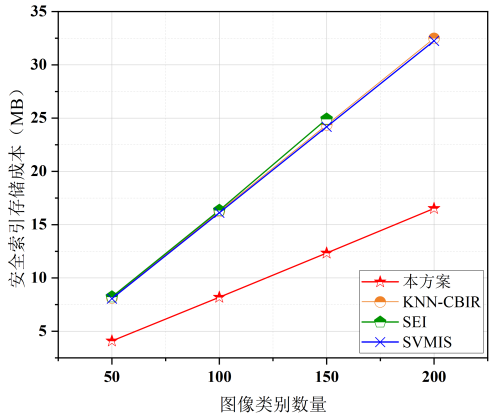


图 **3.10**　多类别图像生成索引的存储成本对比

3.8　本章小结

本章提出了一种基于深度学习的图像可搜索加密方案，该方案使用两段式特征提取模型来提取图像特征向量，使得每张图像的特征向量可用类别标签和图像特征两部分来表示，并使用类别标签构建了基于 K-means 聚类的层次索引树以提高检索效率。为保证索引的安全性，本方案对图像的两段特征采用了不同的加密算法，分别实现了密文域下 $H^{(1)}$ 欧式距离和 $H^{(2)}$ 余弦相似度的计算。最后，对方案进行了安全性分析，并在现有数据集上进行实验，综合检索时间和检索准确度对比，本方案能在多类别图像中实现较高检索效率和检索准确度，且存储成本为同类方案中最小。

第 4 章　图像可搜索加密方案中的数据更新算法

4.1　引言

在检索用户权限被撤销或者图像解密密钥被泄露的情况下，为了保证图像的隐私安全，图像所有者需要对图像集中的所有图像重新加密并上传到云服务器中，这会给图像所有者造成沉重的负担。针对上述情景，本章节提出了一种支持代理重加密的图像更新算法，在图像密钥泄露或检索用户权限撤销的情况下，由图像所有者生成重加密密钥，委托云服务器对密文图像进行重加密，重加密后的图像只能用新分发的私钥解密。同时，本章还对基于深度学习的图像可搜索加密方案中已有类别增加新图像和增加新类别图像两种情况下索引的更新进行了讨论，并通过实验进行了分析。

4.2　问题阐述

4.2.1　设计目标

（1）索引和查询的隐私安全：使用的加密算法需要保护索引和查询的安全性，即云服务器无法从密文索引和查询陷门中获得明文信息。

（2）查询的不可链接性：对于每次查询，即使检索图像相同，生成的查询陷门也应该是不同的。

（3）图像重加密：在检索用户权限被撤销或图像解密密钥被泄露时，在不泄露明文图像信息的前提下，云服务器能利用重加密密钥将密文图像重加密为仅能用新私钥解密的图像。

（4）索引更新：在新增少量图像时，能对索引进行同步更新。

4.2.2　系统框架

本方案的系统框架如图4.1所示，方案共包括三个实体，分别是图像所有者（Image owner）、检索用户（Search user）、云服务器（Cloud server），下面将简要介绍各个实体的主要任务。

图像所有者：图像所有者生成的密钥包含图像加解密密钥、索引加密密钥、陷门生成密钥，其中图像加密密钥和索引加密密钥用于图像所有者生成密图和安全索引，陷门生成密钥和图像解密密钥分发给合法用户用于生成查询陷门和解密检索结果。最后，图像所有者将密图和安全索引上传至云服务器中存储。当检索用户权限被撤销或图像密钥泄露时，图像所有者会重新生成新的公私钥对用于图

像加解密，并将私钥分发给现有检索用户用于图像解密，同时生成图像重加密密钥发送给云服务器，之后图像所有者若要上传图像至云服务器，需要使用新公钥加密图像。

云服务器：云服务器为密文图像和安全索引提供了存储空间，在接收到来自检索用户的查询请求时进行相似度计算，并排序返回相似度最高的 k 张图像给检索用户。在收到来自图像所有者的重加密密钥后，云服务器需要对密文图像进行重加密。

检索用户：利用获得的密钥生成查询陷门上传到云服务器中进行检索，同时可解密云服务器返回的检索结果。收到图像所有者分发的新密钥后，更新图像解密密钥，之后解密图像使用更新的密钥。

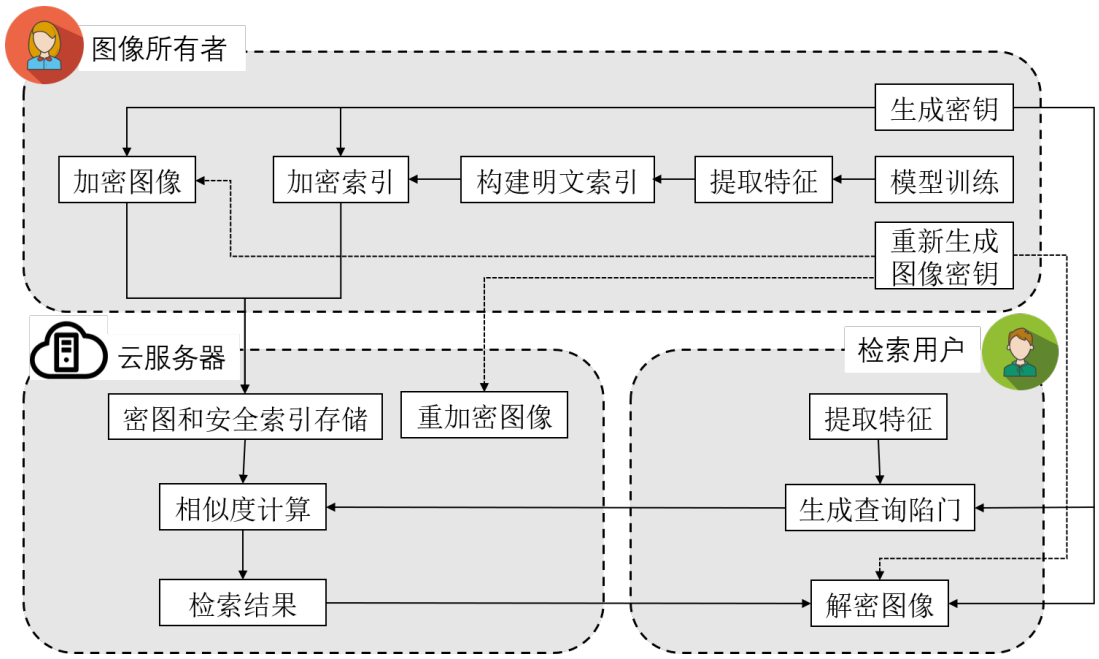


图 **4.1**　系统框架

本方案的算法可分为八个部分：密钥生成（GenKey）、加密图像（EncImage）、生成安全索引（GenIndex）、生成陷门（GenQuery）、检索（Search）、解密结果（EncImage）、重新生成图像密钥（ReGenKey）、图像重加密（ReEnc）。

GenKey：由图像所有者生成密钥集合 ψ_O 和 ψ_U，$\psi_O = \{M_O, M^{-1}, A, U, pk_i, \gamma\}$ 用于索引加密和图像加密，$\psi_U = \{M_U, M, B, sk_i, \gamma\}$ 发送给授权用户以生成查询陷门和解密检索结果。其中密钥 pk_i 用于加密图像，密钥 sk_i 用于解密图像，其余密钥定义与第三章相同，即用于生成安全索引和陷门。

EncImage：图像所有者用密钥 pk_i 加密图像获得密文。

GenIndex：图像所有者构建层次索引树并加密，得到安全索引。

GenQuery：检索用户利用预训练的 CNN 模型提取图像特征并生成查询陷门。

Search：云服务器在密文状态下计算图像之间的相似度，返回 top-k 张图像。

EncImage：检索用户利用私钥解密返回的检索结果。

ReGenKey：当检索用户权限被撤销或图像解密密钥被泄露时，图像所有者生成新的公私钥对 $sk_j = \{a_j\}$，$pk_j = \{g^{a_j}\}$，并将私钥 sk_j 分发给授权用户。此外，图像所有者还需要生成重加密密钥 $rk_{i\to j}$ 发送给云服务器。

ReEnc：云服务收到重加密密钥后，对密图进行重加密运算。

4.3 图像加密和更新算法

Lei 等人提出的方案[65] 支持在不泄露明文数据的情况下将重新加密的任务委托给云服务器，我们将其方案应用于图像加密，在检索用户权限被撤销或者图像解密密钥被泄露时，由图像所有者生成重加密密钥并委托云服务器实现图像数据的代理重加密，重加密后的密文图像仅能用新私钥解密。下面将对图像重加密技术进行详细描述。

（1）初始化：G 是一个阶为 q 的乘法循环群，随机选择 $g \in G$，q 和 g 为公开参数。

（2）密钥生成：图像所有者首先从 Z_q 中随机选择一个元素 a_i，并生成公钥 $pk_i = \{g^{a_i}\}$，私钥 $sk_i = \{a_i\}$，随后将私钥 sk_i 发送给检索用户用于图像解密。

（3）图像加密：设图像明文为 m，图像所有者从 Z_q 中随机选择一个元素 x_i，使用公钥 pk_i 加密明文得到如公式（4.1）中所示为密文 c_i。

$$\begin{aligned} c_i &= (c_{i,1}, c_{i,2}) \\ &= ((pk_i)^{x_i}, mg^{x_i}) \\ &= (g^{a_i x_i}, mg^{x_i}) \end{aligned} \tag{4.1}$$

（4）重加密密钥生成：图像所有者生成新的公私钥对 (sk_j, pk_j)，其中公钥 $sk_j = \{a_j\}$，私钥 $pk_j = \{g^{a_j}\}$，a_j 是随机从 Z_q 中选取的，并生成重加密密钥 $rk_{i\to j}$ 发送给云服务器，$rk_{i\to j} = (rk_{i\to j,1}, rk_{i\to j,2}) = (pk_j^{x_j}, g^{x_j - x_i}) = (g^{a_j x_j}, g^{x_j - x_i})$，其中 $x_j \in Z_q$ 为随机选取。

（5）图像重加密：云服务器收到来自图像所有者的重加密密钥后对图像进行重加密，得到如公式（4.2）所示的重加密图像密文 c_j。

$$\begin{aligned} c_j &= (c_{j,1}, c_{j,2}) \\ &= (rk_{i\to j,1}, c_{i,2} rk_{i\to j,2}) \\ &= (g^{a_j x_j}, mg^{x_i} g^{x_j - x_i}) \\ &= (g^{a_j x_j}, mg^{x_j}) \end{aligned} \tag{4.2}$$

（6）检索用户收到返回的检索结果 c_j 后，利用私钥 $sk_j = \{a_j\}$ 对图像解密，

具体解密过程如公式（4.3）所示。

$$\begin{aligned} m &= \frac{c_{j,2}}{\left(c_{j,1}\right)^{\frac{1}{a_j}}} \\ &= \frac{mg^{x_j}}{\left(g^{a_j x_j}\right)^{\frac{1}{a_j}}} \\ &= \frac{mg^{a_j}}{g^{a_j}} \end{aligned} \tag{4.3}$$

为了更好的理解图像加密和更新的过程，图4.2给出了图像加密和更新算法中的密钥分发示意图。

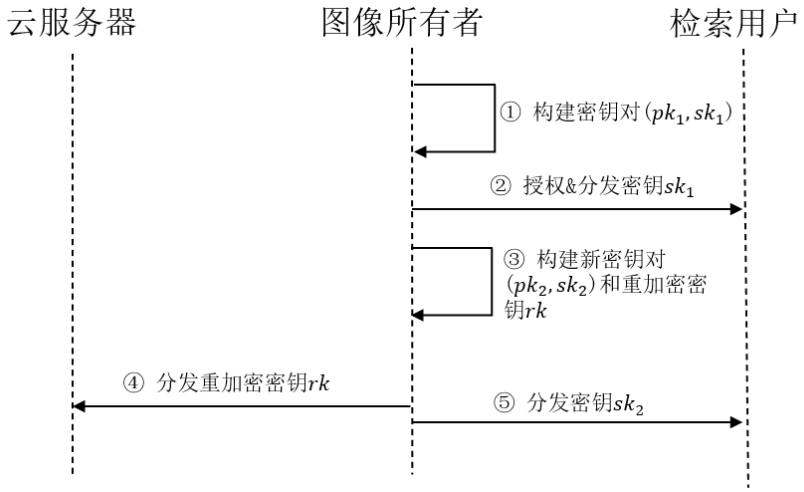


图 **4.2**　图像密钥分发示意图

图像所有者首先生成密钥对 (pk_1, sk_1)，利用公钥 pk_1 加密图像，并给合法授权用户分发私钥 sk_1 用于解密图像。当检索用户权限被撤销或图像密钥被泄露时，图像所有者构建新密钥对 (pk_2, sk_2) 和重加密密钥 $rk_{1\to 2}$，并分发重加密密钥 $rk_{1\to 2}$ 给云服务器，分发图像解密密钥 sk_2 给检索用户。

4.4　索引更新

4.4.1　已有类别增加新图像

在图像所有者需要对已有类别中添加新的图像时，图像所有者首先需要利用预训练的特征提取模型从图像中提取图像特征向量，并对提取的向量进行加密。然后从安全索引的根节点出发，利用 $H^{(1)}$ 查找到正确分类，用 $H^{(2)}$ 对索引进行更新，更新过程如图4.3所示。因为同类别图像的类别标签会非常接近，所以我们忽略对索引中节点的更新。

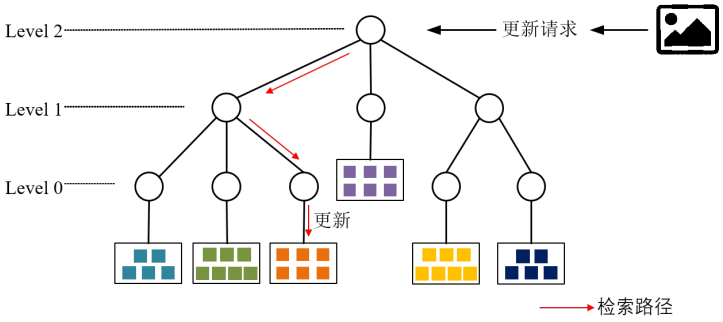


图 **4.3**　更新示意图

下面将对更新的具体步骤进行描述。假设图像所有者有 p 张图像需要更新，首先图像所有者利用预训练的网络模型提取如公式（4.4）所示的图像特征向量。

$$H_i = (H_i^{(1)} || H_i^{(2)}) = (h_{i,1}, h_{i,2}, ..., h_{i,d_1}, f_{i,1}, f_{i,2}, ..., f_{i,d_2}), i \in [1, p] \tag{4.4}$$

如公式（4.5）所示，图像所有者首先将 $H_i^{(1)}$ 扩展为 $2d_1$ 维的向量 $\widetilde{H_i^{(1)}}$。

$$\widetilde{H_i^{(1)}} = (\delta_i h_{i,1}, \delta_i h_{i,2}, ..., \delta_i h_{i,d_1}, \delta_i, \beta_1, \beta_2, ..., \beta_{d_1-1}) \tag{4.5}$$

其中 $\beta_1, \beta_2, ..., \beta_{d_1-1} \in \mathbb{Z}_{p_2}$ 均为随机数，$\delta_i \in \mathbb{Z}_{p_2}$ 为随机正整数，p_1 和 p_2 满足 $p_1 \gg p_2$。

接着，图像所有者利用可逆矩阵 M_U 加密 $\widetilde{H_i^{(1)}}$ 获得密文 $Enc(\widetilde{H_i^{(1)}})$，如公式（4.6）所示，$\vec{\xi_i} \in \mathbb{Z}_{p_2}^{2d_1}$ 是图像所有者随机选择的整数扰动，随机数 $\gamma \in \mathbb{Z}_{p_1}$，且 $\gamma \gg 2|max(\vec{\xi_i})|$。

$$Enc(\widetilde{H_i^{(1)}}) = M_U \times (\gamma \widetilde{H_i^{(1)}}^T + \vec{\xi_i}^T) \tag{4.6}$$

为保护 $H_i^{(2)}$ 的隐私安全，图像所有者首先将 $H_i^{(2)}$ 转换为 $H_i^{(2)'}$，具体计算方式如公式（4.7）所示。

$$H_i^{(2)'} = \frac{1}{\|H_i^{(2)}\|}(f_{i,1}, f_{i,2}, ..., f_{i,d_2}), i \in [1, p] \tag{4.7}$$

接着将 $H_i^{(2)'}$ 转换为 $d \times d$ 维的矩阵 $\overline{H_i^{(2)}}$，如公式（4.8）所示。

$$\overline{H_i^{(2)}} = (\overline{f_{i,j,1}}, \overline{f_{i,j,2}}, ..., \overline{f_{i,j,d}}), i \in [1, p], j \in [1, d] \tag{4.8}$$

然后将 $\overline{H_i^{(2)}}$ 按照公式（4.9）的方式扩展为 $d \times (2d)$ 维的矩阵 $\widetilde{H_i^{(2)}}$。

$$\widetilde{H_i^{(2)}} = (\overline{f_{i,j,1}}, U_{j,1}, \overline{f_{i,j,2}}, U_{j,2}, ..., \overline{f_{i,j,d}}, U_{j,d}), i \in [1, p], j \in [1, d] \tag{4.9}$$

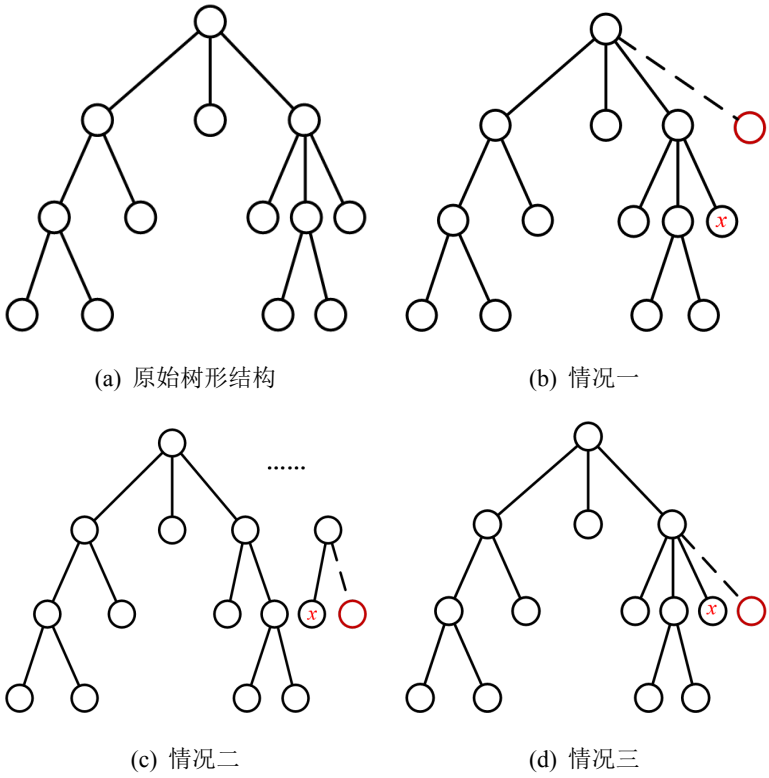
最后，图像所有者使用随机可逆矩阵 M 和随机矩阵 A 加密获得最终密文 $Enc(\widetilde{H_i^{(2)}})$，加密方式见公式（4.10）。

$$Enc(\widetilde{H_i^{(2)}}) = \widetilde{H_i^{(2)}} \times M^{-1} + A \tag{4.10}$$

完成加密操作后，图像所有者将生成的密文提交到云服务器，发送更新请求，由云服务器完成对密文索引的更新。

4.4.2 增加新类别图像

在实际应用中，图像所有者可能还会需要添加某一个分类的图像，这将对索引树的结构造成很大的改变，如图4.4所示，假设原始安全索引结构为4.4(a)，新添加的节点（红色节点）从根节点开始遍历，找到距离最近的叶节点 x，新节点和叶节点 x 的关系可能是兄弟节点或者非兄弟节点，图4.4中的三种情况都会对索引树的上层形态造成影响，最直接的办法是图像所有者重构索引树并加密后上传到云服务器。但是每次增加新的类别都需要图像所有者重新构建层次索引树会造成过多的计算和传输成本。所以本文选择只考虑图4.4(d)这一种情况，即每次查找到距离最近的叶节点 x，并将新添加的节点合并为 x 的兄弟节点，这样做可能和重构的索引树结构不一致，且会降低检索的准确度，但在添加少数类别的情况下，牺牲较少的准确度，减少图像所有者的计算负担是可行的。



(a) 原始树形结构　(b) 情况一

(c) 情况二　(d) 情况三

图 **4.4**　更新情况示例

同时，由于本文提出的方案中类别标签维度与图像类别数量有关，通常类别数量越多，类别标签维度也要增长，这样才能保证类别标签之间的距离。在第三章中，当图像类别数量为 50 时，图像特征向量维度为 24+64 维度，100 类图像时，图像特征向量维度为 32+64 维，为了提供添加图像类别这一操作，在 50 类图像时，本方案选用 32 维的类别标签，即选择 32+64 维的图像特征向量。

当图像所有者需要增加一个类别的时候，首先需要构建该类图像的目标哈希码，并重新训练两段式 CNN 模型，提取该类图像的两段式特征。之后利用该类别图像的 $H^{(1)}$ 求出中心值的代表向量 RC，如公式（4.11）所示，其中 l 代表该类图像的数目。

$$RC = \frac{\sum_{i=1}^{l} H_i^{(1)}}{l} \tag{4.11}$$

接着使用4.4.1小节中的加密方法分别加密 RC 和所有图像的 $H^{(2)}$，最后得到加密的类别中心节点 ERC 和加密的图像特征，再利用上述的方式找到与类别中心节点 ERC 最近的叶节点 x，添加新的分类节点。虽然增加新类别图像时会对索引结构造成改变，但若未发生图像密钥泄露或检索用户权限被撤销的情况，则无需对图像集中的图像进行更新。

4.5 安全性分析

本小节将从索引更新和图像更新两个方面对方案进行安全性分析。

（1）图像安全：图像加密算法的安全性基于离散对数难题 (Discrete logarithm Problem)，即使云服务器已知图像密文、公开参数 g 和 p 以及代理重加密密钥 rk，由于离散对数难题，云服务器仍无法推测出明文图像的信息。因此，本文使用的图像加密算法可以保证图像的安全性。

（2）索引更新安全：在对索引进行上述两种情况的更新时，分别采用了基于 LWE 的安全 kNN 算法和基于随机矩阵的加密算法，其中加密所用的随机可逆矩阵 M_U 和随机矩阵 M 和 A 仅图像所有者拥有，云服务器无法通过密文得到明文信息，具体的安全性证明在第三章中已经给出。因此，本方案可以保证索引更新的安全性。

4.6 实验评估

为了验证上述理论的可行性，本小节对图像加解密和重加密、索引更新等各项指标进行了分析。本实验使用拥有 Windows10 操作系统，3.60GHz 的 AMD，8GB 内存的计算机，使用的语言为 Python。实验选用 Caltech256 图像集，随机抽

取 50 类图像作为基础数据集。分别从图像加密、索引更新（包括已有类别图像更新和增加新图像类别）进行了分析。

4.6.1 图像加密和更新效率分析

本方案采用4.3小节所述的算法加密图像，图4.5给出了图像所有者分别加密 800、1600、2400、3200 和 4000 张图像的时间开销，从图中可知，加密 4000 张图像约花费 20.3828min。同时，本文还计算了云服务器对图像进行重加密的时间开销，如图4.5所示，云服务器重加密 4000 张图像约花费 32.1897min。

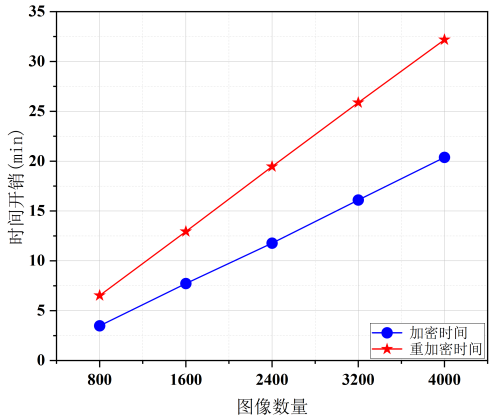


图 **4.5** 图像加密和重加密时间开销

此外，本文还计算了检索用户分别解密 $k=(5,10,20,30,40)$ 张图像时的时间开销，如图4.6所示，检索用户解密 5 张图像约花费 1.3278s。而图像所有者每次生成新的公私钥对以及重加密密钥的时间非常短，可忽略不计。

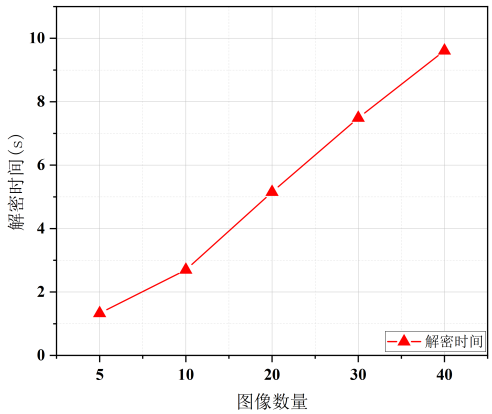
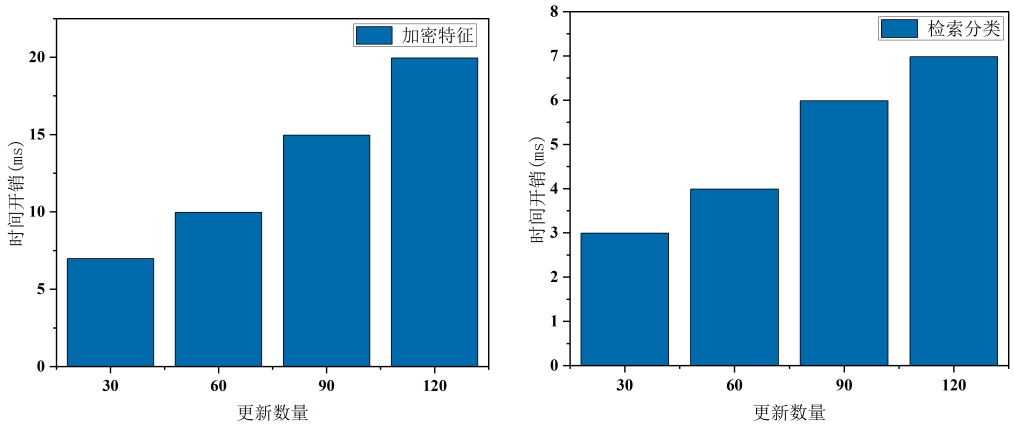


图 **4.6** 图像解密时间开销

4.6.2 索引更新效率分析

针对已有类别增加新图像的情况，主要时间开销包括：提取图像特征、生成加密特征、查询正确分类三部分的时间。通过实验可知，提取 1000 张图像的时间约为 55min，平均每张图像约花费 55s。如图4.7所示，本文给出了新增 30 张、60

张、90 张、120 张图像时，加密图像特征向量和检索到具体分类的时间开销，从图中可以看出，新增 120 张图像时，加密特征向量约花费 19.95ms，检索到具体分类约花费 6.98ms。



(a) 加密特征的时间开销　　(b) 检索分类的时间开销

图 **4.7**　更新多张图像的时间开销

在 50 类图像作为基础图像集时，本文统计了增加 2 类、4 类、6 类、8 类图像时更新索引（包括加密特征和检索到具体位置并更新）所花的时间，如表4.1所示，其中每个类别包含 80 张图像。从表中我们可以看到，新增两类图像的时间开销约为 18.95ms，而在 50 类图像时候构建索引树的时间开销约为 0.53s，在新增少数类别的情况下，通过本文提出的方法更新索引树的时间开销小于重构安全索引树的时间成本。

表 **4.1**　增加不同类别数量图像的时间开销

类别数量	2	4	6	8
时间开销（ms）	18.9509	19.9263	19.9468	20.9153

4.7　本章小结

本章节提出了用于图像更新场景的图像重加密算法，在检索用户权限被撤销或图像解密密钥泄露时，图像所有者只需要重新生成图像加解密密钥和重加密密钥，并委托云服务器完成图像的重加密工作。此外，本文还讨论了实际应用中索引更新的问题，并通过实验对方案进行了评估。

结论

智能手机和数码相机等成像设备的迅速发展使得图像数量快速增长，而本地存储空间和计算资源是有限的。与此同时，云计算的发展为按需访问丰富的计算和存储资源提供了解决方案，这使得许多用户和企业选择将图像数据外包给云服务器并希望之后能对这些图像进行检索。但是图像数据存储在云服务器中，用户就失去了对数据的绝对控制权，一些敏感的隐私图像信息一旦泄露，将会给企业和个人造成巨大的损失。为保护图像的隐私安全，通常会在外包前对图像数据进行加密。虽然明文域的CBIR方案已经非常成熟，但这些方案无法直接应用于密文域，为了解决这个问题，提出了很多基于隐私保护的CBIR方案，这些方案能够在保护图像信息的同时支持密文域下的图像检索。但是现有的基于隐私保护的图像检索方案中存在检索准确度低，检索效率不高以及存储成本较大等问题。本文对图像可搜索加密领域的特征提取算法、图像加密算法、特征向量加密算法、索引结构等方面展开了研究，针对上述问题提出了相应的解决方案。同时，针对实际场景提出了支持代理重加密的图像更新算法，并对索引的更新分情况进行了讨论。

本文的研究内容主要包括以下几个方面：

（1）本文提出了一种基于随机矩阵的特征加密算法，该算法利用随机矩阵的性质消除了随机数对余弦相似度排序的影响，能够保持密文特征向量和查询陷门余弦相似度排序与明文余弦相似度排序一致。同时，本文将该算法应用于图像特征的加密，并通过实验和理论分析证明了该算法能够实现密文域下安全、高效且准确的余弦相似度计算与排序。

（2）本文设计并实现了基于深度学习的图像可搜索加密原型系统。该原型系统通过预训练的CNN模型提取图像特征向量，将每张图像的特征向量表示为类别标签和图像特征两部分，并利用类别标签构建了基于K-means聚类算法的层次索引树以提高检索效率。为保护索引的隐私安全，对类别标签采用基于LWE的安全kNN算法加密，对图像特征采用本文提出的基于随机矩阵的算法加密，两种加密算法分别实现了密文状态下欧式距离和密文状态下余弦相似度的计算，有效的保障了索引和查询向量的隐私安全。最后通过实验对比对系统的性能进行了评估。实验结果表明，本文提出的方案能够在多类别图像中实现较高的检索准确度和检索效率，且存储成本也是同类方案中最低的。

（3）针对实际应用场景中检索用户权限被撤销或图像解密密钥被泄露的情况，本文提出了一种支持代理重加密的图像更新算法。当某个检索用户权限被撤销或者图像解密密钥被泄露时，图像所有者仅需生成新的图像加解密密钥和重加密密

钥，并将重加密密钥发送给云服务器，委托其对密文图像进行重加密。云服务器利用重加密密钥将密文图像重加密为仅能用新私钥解密的图像。通过该算法加密图像，既保证的图像的隐私安全，又将重新加密图像的计算任务交由云服务器完成，有效减少了图像所有者本地的计算负担。同时，本文还对基于深度学习的图像可搜索方案中已有类别增加新图像和增加新类别图像两种情形下安全索引的更新方式进行了讨论，并通过实验进行了分析。

综上所述，本文提出的基于深度学习的图像可搜索方案有效解决了现有密文域 CBIR 方案中存在的检索准确度较低、检索效率不高和存储代价太大等问题。针对实际应用场景，为减轻图像所有者的计算负担，本文提出了一种支持代理重加密的图像数据更新算法，并对索引更新情况进行了讨论。本文的研究内容对于图像可搜索加密领域的研究具有一定的参考意义。

针对本文的主要研究内容，未来的研究工作可以关注以下几个方面：

（1）支持图像数据动态更新的检索方案

本文提出的方案在已知图像类别数量的前提下设计目标哈希码并训练 CNN 网络模型，在论文的第四章对本文方案中索引的更新情况进行了讨论，但是提出的更新方案都牺牲了检索准确度。并且，在图像类别增加时，图像所有者需要设计新的目标哈希码并重新训练 CNN 网络模型。因此，如何在不牺牲检索准确度的前提下，实现支持高效动态更新的安全图像检索方案是未来的研究重点。

（2）支持数据完整性验证的检索方案

在图像外包的情况下，图像所有者失去了对图像的绝对控制权，存储在云服务器中的图像数据可能存在被恶意篡改、伪造或者删除的情况。目前支持数据完整性验证的技术有很多，例如默克尔哈希树（Merkle Hash Tree，MHT）等。如何将现有的数据完整性验证技术应用于图像可搜索加密领域以提高对图像数据的隐私保护是未来的研究内容之一。

（3）支持版权保护的检索方案

在实际使用场景中，检索用户可能会将返回的检索图像恶意转发给其他非法用户，造成图像泄露，图像所有者的版权无法得到保证。因此，有必要对检索图像设计一种跟踪机制来确保图像所有者的版权，常见的图像跟踪技术包括数字水印技术。数字水印技术通过在检索结果中嵌入带有身份信息的水印信息来实现跟踪，当发现可疑副本时，从可疑副本图像中提取代表身份信息的水印跟踪到恶意用户，从而实现对图像的版权保护。未来，实现将版权保护与图像可搜索加密相融合的方案也是重要的研究内容之一。

参考文献

[1] 任奎. 云计算中图像数据处理的隐私保护. 网络与信息安全学报, 2016, 2(1):12–17.

[2] 徐鹏, 林璟锵, 金海, 等. 云数据安全. 北京: 机械工业出版社, 2018.

[3] 51CTO. Snapchat 回应照片外泄事件：责任在第三方应用, (2014-10-11). https://www.51cto.com/article/453887.html.

[4] 中国新闻网. 英国政府承包商 Capita 遭黑客攻击损失额达 2000 万英镑, (2023-05-10). https://www.chinanews.com.cn/gj/2023/05-10/10004964.shtml.

[5] 梅园, 叶登攀, 刘昌瑞. 加密域图像检索技术综述. 华南理工大学学报（自然科学版）, 2018, 46(5):78–86.

[6] Lu W, Swaminathan A, Varna A L, et al. Enabling search over encrypted multimedia databases. In: Proc of Media Forensics and Security, volume 7254. SPIE, 2009, 404–414.

[7] Lu W, Varna A L, Swaminathan A, et al. Secure image retrieval through feature protection. In: Proc of 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2009, 1533–1536.

[8] Zhang Y, Zhuo L, Peng Y, et al. A secure image retrieval method based on homomorphic encryption for cloud computing. In: Proc of 2014 19th International Conference on Digital Signal Processing. IEEE, 2014, 269–274.

[9] Lu W, Varna A L, Wu M. Confidentiality-Preserving Image Search: A Comparative Study Between Homomorphic Encryption and Distance-Preserving Randomization. IEEE Access, 2014, 2:125–141.

[10] Wong W K, Cheung D W l, Kao B, et al. Secure kNN computation on encrypted databases. In: Proc of Proceedings of the 2009 ACM SIGMOD International Conference on Management of data. 2009, 139–152.

[11] 吴颖, 李璇, 金彪, 等. 隐私保护的图像内容检索技术研究综述. 网络与信息安全学报, 2019, 5(4):14–28.

[12] Xia Z, Xiong N N, Vasilakos A V, et al. EPCBIR: An efficient and privacy-preserving content-based image retrieval scheme in cloud computing. Information Sciences, 2017, 387:195–204.

[13] Xia Z, Wang X, Zhang L, et al. A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing. IEEE transactions on information forensics and security, 2016, 11(11):2594–2608.

[14] Yuan J, Yu S, Guo L. SEISA: Secure and efficient encrypted image search with access

control. In: Proc of 2015 IEEE conference on computer communications (INFOCOM). IEEE, 2015, 2083–2091.

[15] Li X, Xue Q, Chuah M C. CASHEIRS: Cloud assisted scalable hierarchical encrypted based image retrieval system. In: Proc of IEEE INFOCOM 2017-IEEE Conference on Computer Communications. IEEE, 2017, 1–9.

[16] 宋甫元, 秦拯, 张吉昕, 等. 基于访问控制安全高效的多用户外包图像检索方案. 网络与信息安全学报, 2021, 7(5):29–39.

[17] Xia Z, Zhu Y, Sun X, et al. Towards Privacy-Preserving Content-Based Image Retrieval in Cloud Computing. IEEE Transactions on Cloud Computing, 2018, 6(1):276–286.

[18] Brakerski Z, Gentry C, Halevi S. Packed ciphertexts in LWE-based homomorphic encryption. In: Proc of Public-Key Cryptography–PKC 2013: 16th International Conference on Practice and Theory in Public-Key Cryptography, Nara, Japan, February 26–March 1, 2013. Proceedings 16. Springer, 2013, 1–13.

[19] Yuan J, Tian Y. Practical Privacy-Preserving MapReduce Based K-Means Clustering Over Large-Scale Dataset. IEEE Transactions on Cloud Computing, 2019, 7(2):568–579.

[20] Shen M, Cheng G, Zhu L, et al. Content-based multi-source encrypted image retrieval in clouds with privacy preservation. Future Generation Computer Systems, 2020, 109:621–632.

[21] Zhu Y, Yu J, Jia C. Initializing k-means clustering using affinity propagation. In: Proc of 2009 Ninth International Conference on Hybrid Intelligent Systems, volume 1. IEEE, 2009, 338–343.

[22] Li Y, Ma J, Miao Y, et al. Similarity Search for Encrypted Images in Secure Cloud Computing. IEEE Transactions on Cloud Computing, 2022, 10(2):1142–1155.

[23] Liu Z, Zhang Y, Ye J. Encrypted Image Retrieval Scheme Based on Lightweight Neural Network. In: Proc of Application of Intelligent Systems in Multi-modal Information Analytics: 2021 International Conference on Multi-modal Information Analytics (MMIA 2021), Volume 1. Springer, 2021, 921–927.

[24] 秦姣华, 黄家华, 向旭宇, 等. 基于卷积神经网络和注意力机制的图像检索. Telecommunication Engineering, 61(3).

[25] Li Y, Ma J, Miao Y, et al. Traceable and controllable encrypted cloud image search in multi-user settings. IEEE Transactions on Cloud Computing, 2020, 10(4):2936–2948.

[26] 李颖莹, 马建峰, 苗银宾. 基于边缘计算的支持多密钥的加密图像检索. 通信学报, 2020, 41(4):14–26.

[27] Yang T, Ma J, Miao Y, et al. MU-TEIR: Traceable Encrypted Image Retrieval in the

Multi-User Setting. IEEE Transactions on Services Computing, 2023, 16(2):1282–1295.

[28] Li Y, Ma J, Miao Y, et al. DVREI: Dynamic Verifiable Retrieval Over Encrypted Images. IEEE Transactions on Computers, 2021, 71(8):1755–1769.

[29] Hsu C Y, Lu C S, Pei S C. Image feature extraction in encrypted domain with privacy-preserving SIFT. IEEE transactions on image processing, 2012, 21(11):4593–4607.

[30] Bai Y, Zhuo L, Cheng B, et al. Surf feature extraction in encrypted domain. In: Proc of 2014 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2014, 1–6.

[31] Bellafqira R, Coatrieux G, Bouslimi D, et al. Content-based image retrieval in homomorphic encryption domain. In: Proc of 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, 2015, 2944–2947.

[32] 陈帆. 量化 SIFT 和同态加密的隐私保护图像检索方法. 传感器与微系统, 2017, 36(5):83–87.

[33] Ferreira B, Rodrigues J, Leitao J, et al. Privacy-preserving content-based image retrieval in the cloud. In: Proc of 2015 IEEE 34th symposium on reliable distributed systems (SRDS). IEEE, 2015, 11–20.

[34] Xia Z, Jiang L, Liu D, et al. BOEW: A Content-Based Image Retrieval Scheme Using Bag-of-Encrypted-Words in Cloud Computing. IEEE Transactions on Services Computing, 2022, 15(1):202–214.

[35] Cheng H, Zhang X, Yu J, et al. Markov Process Based Retrieval for Encrypted JPEG Images. In: Proc of 2015 10th International Conference on Availability, Reliability and Security. 2015, 417–421.

[36] Cheng H, Zhang X, Yu J, et al. Encrypted JPEG image retrieval using block-wise feature comparison. Journal of Visual Communication and Image Representation, 2016, 40:111–117.

[37] Xia Z, Jiang L, Ma X, et al. A Privacy-Preserving Outsourcing Scheme for Image Local Binary Pattern in Secure Industrial Internet of Things. IEEE Transactions on Industrial Informatics, 2020, 16(1):629–638.

[38] Kitayama M, Kiya H. HOG feature extraction from encrypted images for privacy-preserving machine learning. In: Proc of 2019 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia). IEEE, 2019, 80–82.

[39] Qin Z, Yan J, Ren K, et al. Towards efficient privacy-preserving image feature extraction in cloud computing. In: Proc of Proceedings of the 22nd ACM international conference on multimedia. 2014, 497–506.

[40] Jiang M, Yang H. Secure outsourcing algorithm of BTC feature extraction in cloud computing. IEEE Access, 2020, 8:106958–106967.

[41] Chen G, Chen Q, Zhu X, et al. Encrypted image feature extraction by privacy-preserving MFS. In: Proc of 2018 7th International Conference on Digital Home (ICDH). IEEE, 2018, 42–45.

[42] Lowe D G. Distinctive image features from scale-invariant keypoints. International journal of computer vision, 2004, 60:91–110.

[43] Bay H, Tuytelaars T, Van Gool L. Surf: Speeded up robust features. Lecture notes in computer science, 2006, 3951:404–417.

[44] Ojala T, Pietikainen M, Harwood D. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In: Proc of Proceedings of 12th international conference on pattern recognition, volume 1. IEEE, 1994, 582–585.

[45] Ojala T, Pietikäinen M, Harwood D. A comparative study of texture measures with classification based on featured distributions. Pattern recognition, 1996, 29(1):51–59.

[46] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: Proc of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 1. 2005, 886–893.

[47] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, 86(11):2278–2324.

[48] Krizhevsky A, Sutskever I, Hinton G E. ImageNet Classification with Deep Convolutional Neural Networks. In: Proc of Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1. Red Hook, NY, USA: Curran Associates Inc., 2012, 1097–1105.

[49] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[50] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. In: Proc of Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, 770–778.

[51] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks. In: Proc of Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, 4700–4708.

[52] Lu X, Song L, Xie R, et al. Deep binary representation for efficient image retrieval. Advances in Multimedia, 2017, 2017.

[53] Lin K, Yang H F, Hsiao J H, et al. Deep learning of binary hash codes for fast image retrieval. In: Proc of Proceedings of the IEEE conference on computer vision and

pattern recognition workshops. 2015, 27–35.

[54] Li Y, Ma J, Miao Y, et al. Secure and verifiable multikey image search in cloud-assisted edge computing. IEEE Transactions on Industrial Informatics, 2020, 17(8):5348–5359.

[55] Zhu Y, Sun X, Xia Z, et al. Enabling Similarity Search over Encrypted Images in Cloud. Information Technology Journal, 2014, 13:824–831.

[56] 卓力, 龙海霞, 彭远帆, 等. 加密域图像处理综述. 北京工业大学学报, 2016, 42(2):174–183.

[57] Boneh D, Goh E J, Nissim K. Evaluating 2-DNF Formulas on Ciphertexts. In: Proc of TCC, volume 3378. Springer, 2005, 325–341.

[58] Gentry C. A fully homomorphic encryption scheme. Stanford university, 2009.

[59] Zhang Y, Zhuo L, Peng Y, et al. A secure image retrieval method based on homomorphic encryption for cloud computing. In: Proc of 2014 19th International Conference on Digital Signal Processing. IEEE, 2014, 269–274.

[60] Agrawal R, Kiernan J, Srikant R, et al. Order preserving encryption for numeric data. In: Proc of Proceedings of the 2004 ACM SIGMOD international conference on Management of data. 2004, 563–574.

[61] Boldyreva A, Chenette N, Lee Y, et al. Order-Preserving Symmetric Encryption. In: Joux A, (eds.). Proc of Advances in Cryptology - EUROCRYPT 2009. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, 224–241.

[62] 张玉清, 王晓菲, 刘雪峰, 等. 云计算环境安全综述. 软件学报, 2016, 27(6):1328–1348.

[63] Wang X, Ma J, Liu X, et al. Search in my way: Practical outsourced image retrieval framework supporting unshared key. In: Proc of IEEE INFOCOM 2019-IEEE Conference on Computer Communications. IEEE, 2019, 2485–2493.

[64] Griffin G, Holub A, Perona P. Caltech-256 object category dataset. 2007.

[65] Lei L, Cai Q, Chen B, et al. Towards Efficient Re-encryption for Secure Client-Side Deduplication in Public Clouds. In: Lam K Y, Chi C H, Qing S, (eds.). Proc of Information and Communications Security. Cham: Springer International Publishing, 2016, 71–84.