IJNRD.ORG ISSN : 2456-4184

International Research Journal IJNRD Research Through Innovation

INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG

An International Open Access, Peer-reviewed, Refereed Journal

ENHANCING PRIVACY AND ACCURACY IN OUTSOURCED SIFT: EFFICIENT IMAGE FESTURE EXTRACTION

Ramanan K[1], David Prakash M[2], Jai Suriya S[3], Krishna Kumar G[4]
[1]*Assistant professor*, [2] *student*, [3] *student*, [4] *student*
Department of Computer Science and Engineering,
Paavai Engineering College (Autonomous)
Pachal, Namakkal, Tamil Nadu., India.

Abstract : The advent of cloud computing has spurred the demand for secure and efficient picture recovery methods. This paper proposes a novel technique that harnesses the power of cloud infrastructure while prioritizing data security and retrieval accuracy. Central to the approach is the utilization of the Scale Invariant Feature Transformation (SIFT) algorithm for robust feature extraction from images. These features, encapsulating distinctive local image structures, serve as the basis for subsequent retrieval operations. Upon extraction, the system computes the Manhattan distances between the query images SIFT descriptors and those of images stored within the cloud database. This distance metric, known for its effectiveness in feature matching tasks, facilitates the identification of candidate images that closely resemble the query. To optimize retrieval efficiency, a balancing index tree structure is employed for organizing and storing the feature descriptors of images. This ensures rapid search operations, even in scenarios with a vast repository of images. Furthermore, stringent measures are implemented to safeguard data integrity and confidentiality. Before storage in the cloud, images and associated index are stored securely, mitigating risks associated with unauthorized access. The proposed technique offers a comprehensive solution for secure and efficient picture recovery in cloud environments. By integrating advanced image processing algorithms, efficient data structures, and robust encryption mechanisms, the system provides users with a reliable means to retrieve relevant images while safeguarding sensitive information.

Index Terms - Convolutional Neural Networks (CNNs), Scale Invariant Feature Transformation (SIFT), Class Similarity Network (CSN)

INTRODUCTION

The process of finding images that visually resemble a particular image is known as similarity-based image search. This method has certain drawbacks even though it has shown to be helpful in a variety of situations. The quality of the feature extraction and similarity metrics employed determines how accurate similarity-based image search can be. Images that share a similar visual aspect may not always share the same meaning or content. The context of an image is irrelevant when using similarity-based image search. Images that share a similar visual appearance might not share the same message or intended purpose. It is inefficient in managing changes in orientation, illumination, colour, and scale. Images that differ in lighting, colour, scale, orientation, or other aspects but share a similar content may not be recognised as such.
The method of safely searching for photos saved in cloud-based storage systems is called similarity-based image search. Before the photos are uploaded to the cloud, they are encrypted to prevent anyone from accessing them without the

right decryption keys. Using machine learning methods, the images are first transformed into feature vectors in this methodology. After that, these feature vectors are uploaded to the cloud and encrypted using a safe encryption procedure. The query image that the user submits to search for a certain image is likewise transformed into a feature vector and encrypted using the same process. The cloud server receives the encrypted query vector after which it compares it to the encrypted feature vectors of the cloud-stored images.
The cloud server uses a similarity metric to compare the encrypted query vector with the encrypted feature vectors of the images. The similarity metric is designed to preserve privacy by ensuring that the similarity between two encrypted feature vectors is not revealed. The cloud server returns the encrypted feature vectors of the images that are most similar to the encrypted query vector. The client-side application then decrypts the returned encrypted feature vectors to obtain the corresponding images. This technique is useful in scenarios where privacy is a major concern, such as in medical image search or national security applications. By encrypting the images and using similarity-based search, it is possible to ensure that sensitive images are protected while still enabling efficient search and retrieval.

OBJECTIVES

Implement a Privacy-Preserving Image Recognition System: Develop a system capable of recognizing images while ensuring the privacy and security of user data throughout the recognition process.
Integrate the SIFT Algorithm for Feature Extraction: Incorporate the Scale Invariant Feature Transformation (SIFT) algorithm into the system to extract robust features from query images, enabling effective comparison with database images.
Handle Variations in Scale, Rotation, and Illumination: Ensure that the system can handle variations in scale, rotation, and illumination by utilizing the invariant properties of the SIFT algorithm.
Employ Manhattan Distance for Similarity Measurements: Utilize the Manhattan distance metric to compare the features extracted from query images with those of images stored in the database, facilitating efficient and accurate similarity measurements.
Ensure Efficient and Accurate Image Retrieval: Design the system to efficiently retrieve relevant images from the database based on similarity measurements, providing users with accurate results in a timely manner.
Provide User-Friendly Interfaces: Design intuitive and user-friendly interfaces for interacting with the system, allowing users to easily submit query images, view results, and manage their privacy preferences.
Ensure Scalability and Robustness: Build the system to be scalable and robust, capable of handling large volumes of image data and maintaining performance under varying conditions and loads.

RESEARCH METHODOLOGY

[1] SONG, WEIWEI As a Propose a unique deep hashing convolutional neural network (DHCNN) to concurrently retrieve the same pictures and classify their semantic labels in a unified framework. In greater element, a convolutional neural community (CNN) is used to extract high dimensional deep capabilities. Then, a hash layer is perfectly inserted into the community to switch the deep features into compact hash codes. In addition, a fully connected layer with a softmax characteristic is performed on hash layer to generate class distribution. Finally, a loss characteristic is elaborately designed to concurrently remember the label loss of each picture and similarity lack of pairs of features. In extra element, here first undertake a CNN to extract high-dimensional deep functions from raw remote sensing pictures. Then, a hash layer is flawlessly inserted into the CNN to encode the high-dimensional deep functions to low-dimensional hash codes. In addition, a fully linked layer with a softmax function is done on hash layer to generate class distribution. Finally, here elaborately layout a loss characteristic to teach DHCNN, wherein the label data of each photo and similarity facts of pairs of snap shots are simultaneously considered to improve the capability of representation of functions. Once DHCNN is educated enough, for a question picture, can generate its hash code with the aid of binarizing the output of hash layer, then, the retrieval may be without difficulty finished thru Hamming distance ranking. In addition, the semantic labels of snap shots, inclusive of the question photograph and its comparable pictures, can be obtained by using feeding their semantic features into the softmax classifier.

[2] M. V. VISHNUDAS In their study, Implement design of a secure via-design cloud federation platform for the general public sector; here exactly delineate the real data storage integrity that needs of cloud computing environments and the studies inquiries to be tackled to adopt blockchain-primarily based databases. Proposed blockchain-based database pursuits at offering a replicated database whose integrity is testified via good enough evidences stored on a revolutionary designed blockchain device. Namely, right here devise a -layer blockchain that, via the primary-layer, guarantees overall performance and, through a principled exploitation of the second-layer, ensures strong integrity. More specially, the primary-layer employs a lightweight dispensed consensus protocol that assures low latency and excessive throughput. These layer pursuits at quick and reliably storing evidences of every operation completed on a distributed database. However, this deposit provides susceptible data integrity guarantees because of the shortage of PoW. Thus, the second-layer is designed as a PoW-based blockchain that stores evidences of (part of) the database operations logged through the first-layer. These evidences are stored with strong records integrity guarantees but with negative performance. Indeed, the principled interplay among the two layers permits acquiring a typical performance development and effective assurances on facts integrity. Furthermore, a greater thorough and formal examination of the tradeoff among overall performance and records integrity guarantees is needed to show the efficacy of proposed design towards the diagnosed threats.

Finally, learning on the feasibility of realising extra solid blockchains is essential to allow their huge adoption as dependable garage infrastructures, e.g. Inside the context of cloud computing environments.

[3] ZHENGZHUO HAN In this paper we Propose a novel model referred to as class-similarity network (CSN), which targets for image category and similarity prediction at the identical time. In order to further improve performance, hear build and train CSNs, and varieties of facts from them, i.e., deep capabilities and similarity rankings, are consolidated to degree the very last similarity among pictures. Besides, the optimum fusion theorem in biometric authentication, which offers a theoretical scheme to ensure that fusion will virtually result in a better overall performance, is used to behavior score fusion. CSN is composed of two category branches, which percentage precisely the identical structure as well as the equal weights, and one similarity learning network. The inputs to CSN should be a couple of pix, each category department targets to categorise the corresponding image. Suppose there are n instructions, then the remaining layer of the primary (or second) type branch outputs an n-duration vector p (or p'), which suggests the expected possibilities that the enter image relates to every magnificence. Moreover, here talk over with the activation vector "finding in" a category department's penultimate layer as the feature vector of the input photo. Besides, the tradeoff parameter λ has been tentatively selected, eating plenty time; it might be robotically found out as a substitute. Furthermore, because the category branches of CSNs are primarily based on GoogLeNet and ResNet50, which have been each pretrained on regular RGB pixels, this proposed strategies are confirmed on optical HRRS pictures that have 3 channels, similar to regular pictures.

[4] WEIXUN ZHOU In this paper, we Implement a novel multilabel RSIR method based on fully convolutional network (FCN), which has numerous blessings over the existing widespread multilabel RSIR techniques. In trendy, general techniques inclusive of comprise a couple of steps inclusive of image segmentation, vicinity characteristic extraction, label annotation, and so forth., at the same time as proposed technique integrates these steps in a unmarried framework based on FCN which simplifies the technique of multilabel analysis and feature extraction from image. More importantly, fashionable methods need to fuse distinct handmade capabilities to be able to acquire exact performance, while this proposed method can analyze CNN features which might be easy but powerful for multilabel RSIR troubles. In this technique, first teach a FCN community tailored from the pre-trained CNN to generate semantic segmentation map of every picture within the considered archive. Here then compute multilabel vector and extract vicinity convolutional functions (RCFs) of every image based on corresponding segmentation map. For the computation of multilabel vector, it could be without delay obtained from the segmentation map, while for the extraction of RCFs, feature map upsampling and flattening, and region based search are hard to find the nearby convolutional features that lie in each related location. Finally apply the extracted RCFs to carry out multilabel retrieval the usage of a place-based similarity measure. In this approach, FCN is first educated to expect the segmentation map of each photo in proposed retrieval archive. Then achieve multilabel vectors and extract unmarried-scale and multiscale RCFs of each image based on corresponding segmentation map. Finally, the location-based distance is used to measure the similarity between query picture and different pictures within the archive.

PROBLEM DEFINITION

The challenge at hand involves the development of a Privacy-Preserving Image Recognition (PIR) system that can accurately identify images while ensuring the privacy of user data throughout the recognition process. This entails integrating advanced image processing algorithms, notably the Scale Invariant Feature Transformation (SIFT), to extract robust features from query images and compare them with images stored in the database. A critical aspect of this endeavor is accommodating variations in scale, rotation, and illumination to ensure accurate recognition across diverse image datasets. A significant hurdle lies in devising efficient similarity measurement techniques, such as employing the Manhattan distance metric, to effectively compare feature descriptors extracted from query images to those of database images. Moreover, safeguarding user privacy is paramount, necessitating the implementation of robust privacy-preserving techniques, including encryption and anonymization, to protect user data and sensitive image information from unauthorized access. The objective of this project is to develop a scalable and robust PIR system that integrates the SIFT algorithm for feature extraction, employs the Manhattan distance metric for efficient similarity measurements, and implements privacy-preserving techniques to safeguard user privacy. By addressing these challenges, the aim is to create a dependable and secure PIR system that delivers accurate image recognition results while upholding user privacy, thereby meeting the increasing demand for privacy-preserving solutions in image recognition applications.

OVERVIEW OF THE PROJECT

This project aims to develop a Privacy-Preserving Image Recognition (PIR) system that combines advanced image processing techniques with robust privacy-preserving measures. The system will utilize the Scale Invariant Feature Transformation (SIFT) algorithm for feature extraction, enabling the extraction of distinctive and invariant features from query images while accommodating variations in scale, rotation, and illumination. These features will then be compared with those of images stored in the database using the Manhattan distance metric, facilitating efficient similarity measurements. To ensure the privacy of user data and sensitive image information, the system will implement encryption and anonymization techniques. The ultimate goal is to create a scalable and reliable PIR system that delivers accurate image recognition results while safeguarding user privacy, thus addressing the growing need for privacy-preserving solutions in image recognition applications. Through thorough performance evaluations and user-centric design, the project aims to provide a seamless and secure image recognition experience for users across various domains and applications.

SOFTWARE DESCRIPTION

FRONT END: PYTHON

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales. In July 2018, Van Rossum stepped down as the leader in the language community. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library. Python interpreters are available for many operating systems. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of Python's other implementations. Python and CPython are managed by the non-profit Python Software Foundation. Rather than having all of its functionality built into its core, Python was designed to be highly extensible. This compact modularity has made it particularly popular as a means of adding programmable interfaces to existing applications. Van Rossum's vision of a small core language with a large standard library and easily extensible interpreter stemmed from his frustrations with ABC, which espoused the opposite approach. While offering choice in coding methodology, the Python philosophy rejects exuberant syntax (such as that of Perl) in favor of a simpler, less-cluttered grammar. As Alex Martelli put it: "To describe something as 'clever' is not considered a compliment in the Python culture. "Python's philosophy rejects the Perl "there is more than one way to do it" approach to language design in favour of "there should be one—and preferably only one—obvious way to do it".
Python's developers strive to avoid premature optimization, and reject patches to non-critical parts of CPython that would offer marginal increases in speed at the cost of clarity.[When speed is important, a Python programmer can move time-critical functions to extension modules written in languages such as C, or use PyPy, a just-in-time compiler. CPython is also available, which translates a Python script into C and makes direct C-level API calls into the Python interpreter. An important goal of Python's developers is keeping it fun to use. This is reflected in the language's name a tribute to the British comedy group Monty Python and in occasionally playful approaches to tutorials and reference materials, such as examples that refer to spam and eggs (from a famous Monty Python sketch) instead of the standard for and bar.
A common neologism in the Python community is pythonic, which can have a wide range of meanings related to program style. To say that code is pythonic is to say that it uses Python idioms well, that it is natural or shows fluency in the language, that it conforms with Python's minimalist philosophy and emphasis on readability. In contrast, code that is difficult to understand or reads like a rough transcription from another programming language is called unpythonic. Users and admirers of Python, especially those considered knowledgeable or experienced, are often referred to as Pythonists, Pythonistas, and Pythoneers. Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed. Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.
Python's initial development was spearheaded by Guido van Rossum in the late 1980s. Today, it is developed by the Python Software Foundation. Because Python is a multiparadigm language, Python programmers can accomplish their tasks using different styles of programming: object oriented, imperative, functional or reflective. Python can be used in Web development, numeric programming, game development, serial port access and more.

There are two attributes that make development time in Python faster than in other programming languages:

1. Python is an interpreted language, which precludes the need to compile code before executing a program because Python does the compilation in the background. Because Python is a high-level programming language, it abstracts many sophisticated details from the programming code. Python focuses so much on this abstraction that its code can be understood by most novice programmers.
2. Python code tends to be shorter than comparable codes. Although Python offers fast development times, it lags slightly in terms of execution time. Compared to fully compiling languages like C and C++, Python programs execute slower. Of course, with the processing speeds of computers these days, the speed differences are usually only observed in benchmarking tests, not in real-world operations. In most cases, Python is already included in Linux distributions and Mac OS X machines.

BACK END: MySQL

MySQL is the world's most used open source relational database management system (RDBMS) as of 2008 that run as a server providing multi-user access to a number of databases. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack—LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases include: TYPO3, Joomla, Word Press, phpBB, MyBB, Drupal and other software built on the LAMP software stack. MySQL is also used in many high-profile, large-scale World Wide Web products, including Wikipedia, Google(though not for searches), ImagebookTwitter, Flickr, Nokia.com, and YouTube.

Interimages

MySQL is primarily an RDBMS and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or use MySQL "front-ends", desktop software and web applications that create and manage MySQL databases, build database structures, back up data, inspect status, and work with data records. The official set of MySQL front-end tools, MySQL Workbench is actively developed by Oracle, and is freely available for use.

RDBMS Terminology

Before we proceed to explain the MySQL database system, let us revise a few definitions related to the database.

- **Database** − A database is a collection of tables, with related data.
- **Table** − A table is a matrix with data. A table in a database looks like a simple spreadsheet.
- **Column** − One column (data element) contains data of one and the same kind, for example the column postcode.
- **Row** − A row (= tuple, entry or record) is a group of related data, for example the data of one subscription.
- **Redundancy** − Storing data twice, redundantly to make the system faster.
- **Primary Key** − A primary key is unique. A key value cannot occur twice in one table. With a key, you can only find one row.
- **Foreign Key** − A foreign key is the linking pin between two tables.
- **Compound Key** − A compound key (composite key) is a key that consists of multiple columns, because one column is not sufficiently unique.
- **Index** − An index in a database resembles an index at the back of a book.
- **Referential Integrity** − Referential Integrity makes sure that a foreign key value always points to an existing row.

MySQL Database

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons −

- MySQL is released under an open-source license. So you have nothing to pay to use it.
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).
- MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

HTML

HTML is a markup language for describing web documents (web pages).

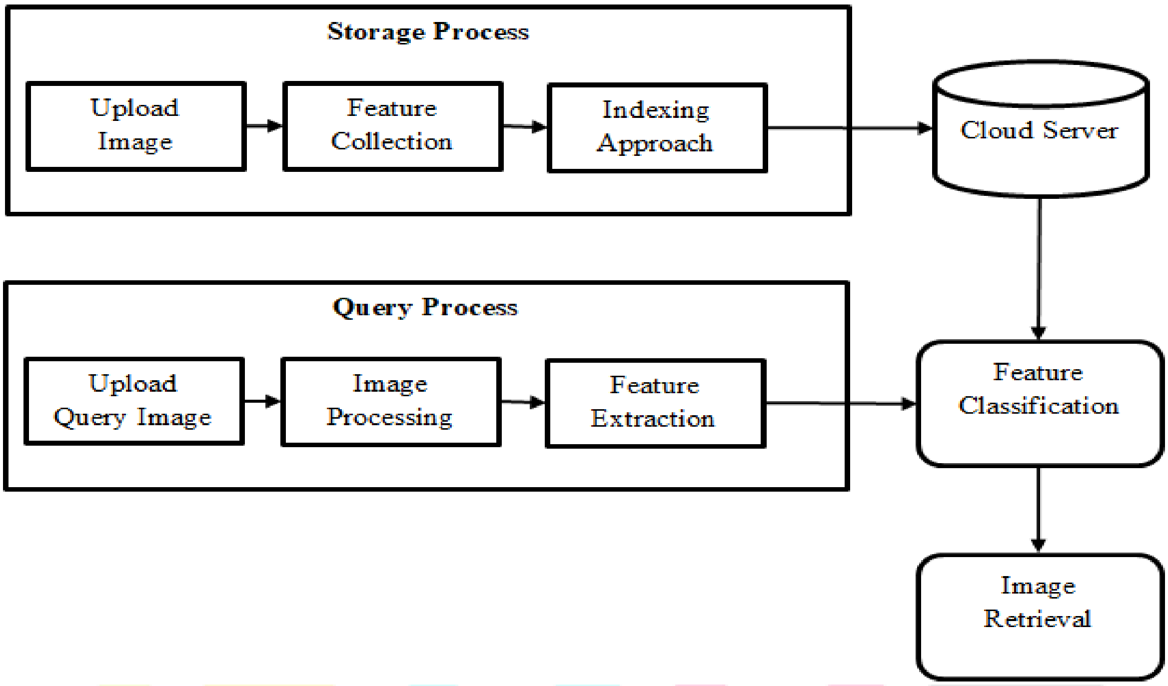
- Hyper is the opposite of linear. It used to be that computer programs had to move in a linear fashion. This before this, this before this, and so on. HTML does not hold to that pattern and allows the person viewing the World Wide Web page to go anywhere, anytime they want.
- Text is what you will use. Real, honest to goodness English letters.

- Mark up is what you will do. You will write in plain English and then mark up what you wrote. More to come on that in the next Primer.
- Language because they needed something that started with " L " to finish HTML and Hypertext Markup Louie didn't flow correctly. Because it's a language, really but the language is plain English.

HTML remains for Hyper Text Markup Language. It is a basic content designing dialect used to make hypertext records. It is a stage free dialect not at all like most other programming dialect. HTML is impartial and can be utilized on numerous stage or desktop. It is this component of HTML that makes it mainstream as standard on the WWW.
This adaptable dialect permits the making of hypertext connections, otherwise called hyperlinks. These hyperlinks can be utilized to unite reports on diverse machine, on the same system or on an alternate system, or can even indicate purpose of content in the same record. HTML is utilized for making archives where the accentuation is on the presence of the record. It is likewise utilized for DTP. The records made utilizing HTML can have content with diverse sizes, weights and hues. It can also contain graphics to make the document more effective.

SYSTEM ARCHITECTURE

A system architecture or systems architecture is the conceptual model that defines the structure, behaviour, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviours of the system. System architecture can comprise system components, the externally visible properties of those components, the relationships (e.g. the behaviour) between them. It can provide a plan from which products can be procured, and systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture; collectively these are called architecture description languages (ADLs).



CONCLUSION

Similarity-based image retrieval in the cloud using query image processing, SIFT feature extraction, and Manhattan distance calculation provides an effective way to search and retrieve visually similar images. The combination of these techniques allows for efficient indexing and retrieval, making it possible to handle large-scale image databases. By leveraging cloud computing resources, the system can scale to accommodate a growing number of images and users. As image collections continue to expand, similarity-based image retrieval in the cloud becomes an invaluable tool for various applications, including image search engines, content-based image retrieval systems, and visual recommendation systems.

I. Acknowledgment

We are grateful to Mr. K. RAMANAN, Assistant Professor, CSE Department, Paavai Engineering College (Autonomous) for mentoring us to present this paper successfully

References

[1] Gao, Peng, Hanlin Zhang, Jia Yu, Jie Lin, Xiaopeng Wang, Ming Yang, and Fanyu Kong. "Secure cloud-aided object recognition on hyperspectral remote sensing images." IEEE Internet of Things Journal 8, no. 5 (2020): 3287-3299.

[2] Shen, Meng, Guohua Cheng, Liehuang Zhu, Xiaojiang Du, and Jiankun Hu. "Content-based multi-source encrypted image retrieval in clouds with privacy preservation." Future Generation Computer Systems 109 (2020): 621-632.

[3] Wu, Tong, Guomin Yang, Yi Mu, Rongmao Chen, and Shengmin Xu. "Privacy-enhanced remote data integrity checking with updatable timestamp." Information Sciences 527 (2020): 210-226.

[4] Fu, Anmin, Shui Yu, Yuqing Zhang, Huaqun Wang, and Chanying Huang. "NPP: A new privacy-aware public auditing scheme for cloud data sharing with group users." IEEE Transactions on Big Data 8, no. 1 (2017): 14-24.

[5] Song, Weiwei, Shutao Li, and Jón Atli Benediktsson. "Deep hashing learning for visual and semantic retrieval of remote sensing images." IEEE Transactions on Geoscience and Remote Sensing 59, no. 11 (2020): 9661-9672.