2016 Sixth International Conference on Instrumentation & Measurement, Computer, Communication and Control

Design of a Cloud Robotics Visual Platform

YIN Lei, ZHOU Fengyu*, Wang Yugang,
YUAN Xianfeng, ZHAO Yang
School of Control Science and Engineering
Shandong University, Jinan, China
E-mail: zhoufengyu@sdu.edu.cn

CHEN Zhumin
School of Computer Science and technology
Shandong University, Jinan, China

***Abstract*—In recent years, the vision of robot has got great development. Because of the low sampling rate and the heavy calculation burden, the vision is obstructed to apply for robot. In order to solve the above problem, in this paper, we present the Cloud Robotics Visual Platform(CRVP) for the development of robot vision. The main contributions of the paper are as follows. (1) The parallel computational model of Map/Reduce is used to decrease the image time cycle of image training and recognition. (2) The image recognition engine is built with the Service-Oriented Architecture. (3) The H264 encoding algorithm and the Real time protocol are adopted to realize the video transmitting. Finally, the experiment is done using ECS of Aliyun. The results show that the platform could effectively improve the robot image recognition cycle and offload the computing load.**

***Keywords*—cloud robot; map/reduce; real time protocol; visual platform**

I. INTRODUCTION

In recent years, the robot has been developing rapidly. The problems such as the single function, poor autonomy, low level of intelligence and high-cost block the service robot in development and application. With the rapid development of cloud technology, the combination of cloud computing technology and robotics would be an important method to solve the above problems.

In 2010, Rajesh Arumugam[1] presented a cloud platform “DAvinCi” for server robot. The platform was based on Robot Operate System(ROS), distributed system architecture of hadoop and the Cloud computing model of Map/Reduce. DAvinci provided the proxy service for robot with binding the server cluster of hadoop and the Ros. Hadoop server cluster contains a number of computing nodes and storage nodes which can finish parallel computing, solve the storage tasks, greatly improve the program execution algorithm efficiency and offload the computation load of robot.

In 2011, in order to solve the resource sharing problem, Zhihui Du [2] built the robot cloud computing center(RCC). The RCC were composed of RCP(Robot Cloud Panel), SB(Sever Broker), RCU(Robot Cloud Units) and MP(Map Layer). The RCP merged the scattered services according to user's requirements and sent the robot serve to robot, and it scheduled the request of the robot, manged the robot's message mapping, monitored the robot state and operates the server agent. The RCC provided the information sharing platform to improve the multi-robot task management coordination capacity and to eliminate incompatible problem between heterogeneous robot and platform.

In 2013, scientists in Europe presented the world wide web for robot named “Roboearth”[3][4]. The goal of Roboearth was to establish a huge open source network database for robot to use internet. All robots around the world could connect the database and update the parameter. Roboearth provided semantic mapping information such as the robot motion sequence, object model and runtime to assist robot to complete the task.

In 2015, Mohanarajah G[5] presented the cloud robotics platform “Raptuya” based Roboearth. The Raptuya was a middleware like Service-Oriented Architecture(SOA). In addition to provide the connection between robot and roboearth, Raptuya optimized the task management, data structure and communication protocol.

Given above all, this paper presents a Cloud Robotics Visual Platform based on Elastic Compute Service(ECS) of Aliyun, establishes the Hadoop Server Cluster, Image Recognition Engine and Video Transmission module. The platform improves the training speed of image gallery through Map/Reduce parallel computational model and encapsulates the image recognition engine to solve the problems the heterogeneous robots call for service. A brief introduction about development and actuality of cloud robotics platform is described in Section 1. The architecture of Cloud Robot Visual Platform is described in Section 2 along with hadoop Distributed Cluster, Image Recognition Engine and Video Transmission module in Sections 3, 4 and 5, respectively. Then our experiment results show that the platform presented could satisfy the request of robot. Finally, discussions about future work and research direction towards completing Cloud Robotics Visual Platform are given.

II. FRAMEWORK OF CRVP

The CRVP contains hadoop distributed server cluster, image recognition engine and video transmission as shown in fig.1. The hadoop distributed server clusters provide the cloud computing and file storage system. Image recognition engine provides the image processing and recognition algorithm. The video transmission contains the video compression and transmission using FFmpeg encoder and Real-time Transport Protocol(RTP), respectively.

Now, many internet companies have developed their own cloud platform. Microsoft presents the “windows Azure” cloud

978-1-5090-1195-7/16 $31.00 © 2016 IEEE
DOI 10.1109/IMCCC.2016.252

platform. Amazon develops the cloud platform "AWS". Alibaba launches the "Aliyun". The advantages of the cloud platform are that it does not require users to maintain the hardware equipment, release the resource of compute and storage when we don't need the cloud. The companies provide the custom network topology of cloud resources and solve the problem of IP layout, load balancing, VPC Interworking, dedicated access. The architecture of those cloud platforms is platform as a service (PaaS). In this paper, we use the ECS of Aliyun to build the CRVP. The ECS interface of Aliyun is shown in fig.2.

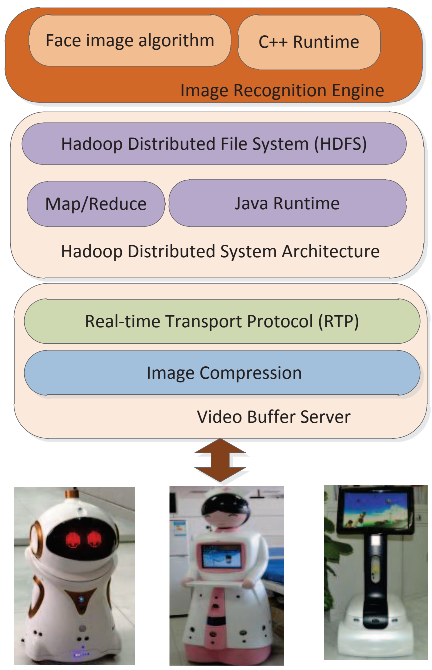


Fig. 1: The Framework of Cloud Robot Visual Platform

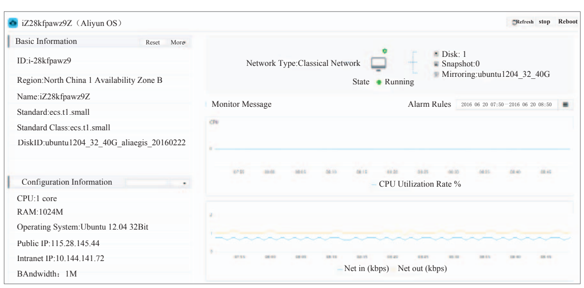


Fig. 2: ECS Interface of Aliyun

The main advantage of distributed cluster is parallel computing. Parallel computing is to accelerate the computation process with more computing resource in the parallel way. It is the method by adding the computing resource to solve the complex problem. The traditional parallel computing used high-performance computer, but the high-performance computer updates so fast and has the disadvantage of high cost. The cloud computing infrastructure depends on cheap computer and realizes the cloud computing by increasing the number of computing nodes. Because of the virtualization technology, node number is free to customize. It is helpful to overcome the waste of resource and improve the flexibility of system resource.

In this paper, Map/Reduce parallel computational model is used to implement the face recognition and training the parameter. The image recognition service is provided for heterogeneous robot. The video transmission module uses the FFmpeg encoder and Real Time Protocol to realize the video encoding and video transmission.

III. DISTRIBUTED SERVER CLUSTER

The NameNode is the main node of cluster and stores the metadata of document such as document name, document directory structure, document properties (generation time, replications, access permissions) and the block list of document in DataNode[6]. NameNode manages the globe document. When robots read the document, the NameNode schedules the user to read the nearest copy file to decrease the bandwidth consumption and access time. The NameNode globally manages the copy of data block and receives the heartbeats and block reports from the DataNodes. The feedback represents the DataNode is operating well. Block reports contain the block list and the storage document information of DataNode. The heartbeats contain the command that the NameNode sent to DataName. DataNodes storage the document data and checksum of block data. The Hadoop Architecture and Information Flow are shown in fig.3.

The images from the image library are uploaded to the DataNode of HDFS[7]. Every slave server preprocesses the images. Then the slaver servers train the parameters of face recognition model. When the robots send the real-time image frame to the cloud platform, the master server preprocesses the image and sends to the slaver server, schedules the slaver server to execute the map (The parallel algorithm of face recognition) task. When the slaver finishes the map task, the DataNode that NameNode specified as reducer calculates the combined results (recognition result). The slaver servers send the block information that storage the result of result to NameNode through the block reports. The master server gets the result and feedbacks to the robot to complete the facial recognition process.

IV. IMAGE RECOGNITION ENGINE

Engine is a kind of application form in internet or programming. The traditional programming experiences procedure-oriented programming, object-oriented programming and service-oriented programming. The CRVP is established based service oriented architecture. The code, the algorithm, the resource and the object are encapsulated into service to provide the computing or recognizing service for heterogeneous robots or service demanders. In this paper, the image recognition engine is presented. Face recognition training and recognition algorithm runs in the C++ environment. Map/Reduce parallel computation of hadoop runs in java environment.

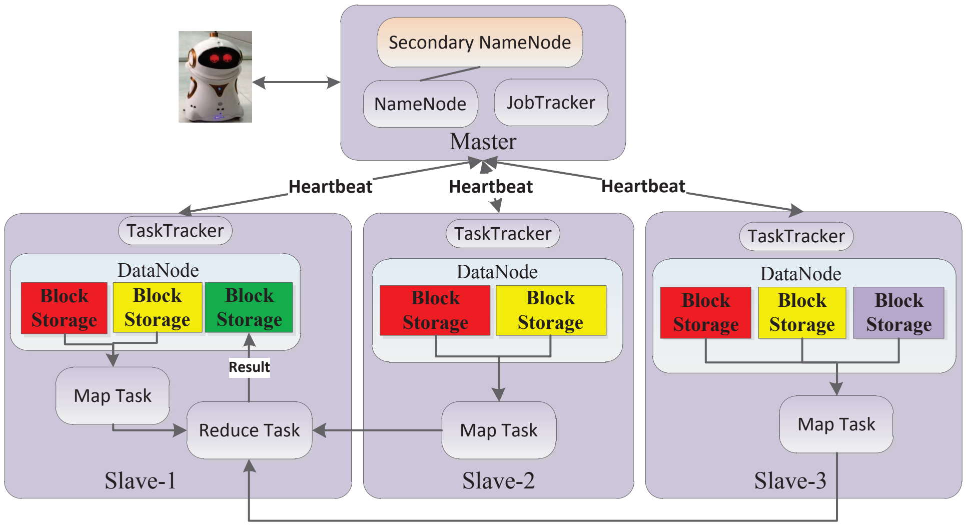


Fig. 3: Hadoop Architecture and Information Flow

We adopt eigenfaces algorithm for training and recognition as shown in [8][9]. The algorithm is shown as follow.

Firstly, a batch of face image is transformed into a set of feature vector called "eigenfaces". Those are the basic module to train the image sets. The recognize process is to project the new image to the subspace of eigenfaces. The subpoint and the length of the projection line is the basis of recognition and judgement.

The original image is projected onto the feature space. The image X is saved as dimensional vector.

$$x_i = [x_1^i, x_2^i, \dots, x_n^i]^T \quad (1)$$

The training set is p, and Matrix X[n][p] is formed, In which, the row represents pixel, and the rank represents each face image. The training sample set of face image minus the average face image, compute the discrete difference, centralize training images.

$$\bar{x}^i = x^i - \mu \quad (2)$$

Now,

$$\mu = \frac{1}{p}\sum_{i=1}^{p} x^i \quad (3)$$

The images centralized form a matrix X with size $n \times p$.

$$\bar{x} = [\bar{x}^1, \bar{x}^2, \dots, \bar{x}^p] \quad (4)$$

Matrix X multiply its transpose of matrix to get the covariance matrix Ω.

$$\Omega = \bar{x}\bar{x}^T \quad (5)$$

Slove the nonzero eigenvalues and the feature vector of the covariance matrix Ω .In normal conditions, training image number p is far less than the image pixel n. The matrix Ω have a maximum of P vector corresponding with nonzero eigenvalue.

$$k \le p \quad (6)$$

The feature vector with the largest maximum eigenvalue responses the biggest difference among the training images. All of the feature vectors with nonzero eigenvalue form the feature space (Eigenfaces Space).

$$\Lambda U = U\Omega \quad (7)$$

In which, U is the feature space.

$$U = [u_1, u_2, \dots, u_k] \quad (8)$$

Each face image can be projected onto the subspace and correspond to a point in space. Each point of subspace corresponds to a face image. We train the image sets to get the eigenface subspace.

Any image centralized could be projected to eigenfaces subspace and get set of coordinate coefficient with the following formula.

$$\bar{x}^i = U\bar{x}^i \quad (9)$$

A vector is composed of M weight vectors, and it is the represent of the new face based eigenface.

$$\Omega^T = [\bar{x}^1, \bar{x}^2, \dots, \bar{x}^M] \quad (10)$$

A face is recognized by the the following formula.

$$\psi_k = \|\Omega - \Omega_k\|^2 \quad (11)$$

Where, Ω is the face need to be recognized, and Ω_k is the training library of faces. We can set the threshold value of ψ_k to predicate the image.

V. VIDEO TRANSMISSION

The advantage of Cloud robot is to offload the image processing and computing task. The key figure to achieve this goal is to transmit the real-time video to cloud platform. In this paper, The FFmpeg encoder[10] and the real time protocol is used to realize the video transmission. The main process contains video encode, video decode, RTP packaging and transmission. Video Transmission process is shown as fig.4.

Firstly the encoder should be initialized and opened to wait the real-time video. Then we add the video stream, set up the corresponding memory space, convert the RGB sets to YUV sets, encode the YUV sets to H264 stream, package the stream according to RTP protocol and send to the image recognition engine.

In this paper, RTP is used as the video transmit protocol. RTP is an application layer protocol for transmitting latency-sensitive data such as video and audio over the Internet. It is the foundation of many Voice-over-IP and media streaming systems. RTP uses a separate channel for monitoring and adjusting the data delivery.

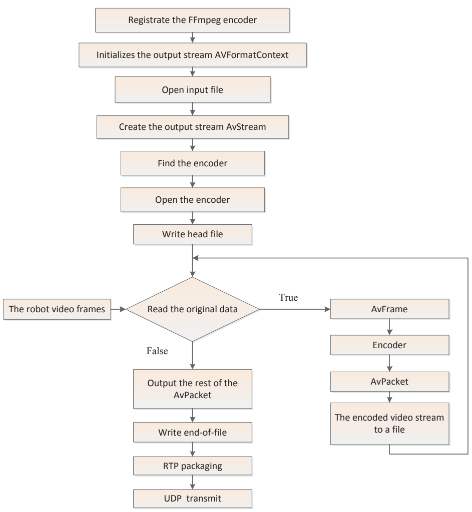


Fig. 4: Video Transmission Process

Real-Time Image Transmission Protocol (RTP)

RTP as shown in fig.5 can be regarded as format. This format is commonly used for TS stream of streaming media [11][12]. Control protocol adopts the RTSP protocol. Timestamp reflects the sampling time of first 8 bits groups. The receiver computes the delay and delay jitter with the timestamp and Checks the packet loss using the serial number.

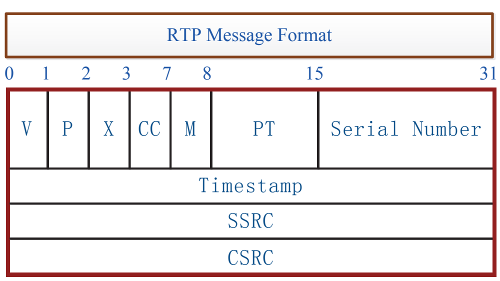


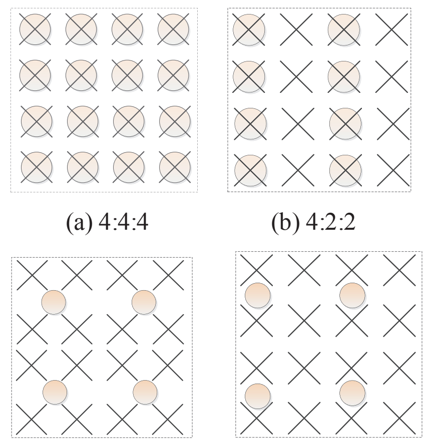
Fig. 5: Real Time Protocol

When the video is transmitted and the length of the NALU exceeds MTU, NALU unit must be subcontracted to fragmentation units (FUs). In this paper, UDP is as support layer protocol. Although the TCP is more reliable than UDP, the delay caused by the TCP linking cannot satisfy the requirements of transmission system. The disadvantage of UDP is unreliable and the data size can't exceed 64 KB once.

YUV Sets

Because of the input of FFmpeg encoder is YUV sets, we need to convert the RGB sets.

Firstly we convert the RGB sets to BGR sets through the channel transformation. All the sampling is based on BGR sets. YUV is a pixel format that separates the brightness parameter and chromaticity parameter. The advantage is to avoid the mutual interference between above two parameters and decrease the chromaticity of sampling rate without affecting the image quality. According to the arrangement of Y, U and V, YUV can be divided into 4:4:4, 4:2:2, 4:2:0 and 4:1:1. Four sampling grid is shown in fig.6. The light sample is represented with a fork and the color sample is expressed in circle.



(a) 4:4:4 (b) 4:2:2

(c) 4:2:0 MPEG-1 (d) 4:2:0 MPEG-2

Fig. 6: Four Sampling Grid

There are two common forms of change of 4:2:0 forms. One form is used for mpeg-2 video as shown in fig.6(d), another kind of form for mpeg-1 as shown in fig.6(c) and ITU-T recommendations. In this paper, YUV420 is adopted for coding input format.

VI. EXPERIMENT

In this paper, Three Elastic Compute Services(ECS. configured as one core, 1G RAM, 1M bandwidth. System version is Ubuntu12.04) of Aliyun are used as the server platform. Java runtime is for the Map/Reduce parallel computing and C++ runtime for image recognition and video transmission. Hadoop version is 1.2.0. Open Source Computer Vision Library is opencv2.4.6. We use the 500 face images from the Library of California institute of technology.

Firstly upload the images as shown in fig.7(a) to the DataNode of HDFS and parallelly preprocess the original images with Map/Reduce. Face detection was carried out on the picture and the face region is cut as shown in fig.7(b). The image will be grayed, normalized to 320×320 as shown in fig.7(c).

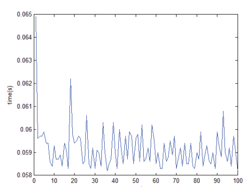
(a) Original images

（b）The image after cutting

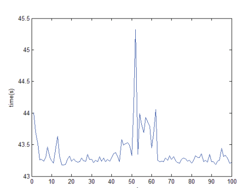
（c）Images after normalization processing

Fig. 7: Faces Preprocess

Then start parallel training with the model of Map/Reduce and the image recognition engine. When robots have sent the real-time face image to the platform, the NameNode will start the distributed computing program and specify the node to run Map/Reduce to finish the image recognition tasks.



(a) Recognition Time



(b) Training time

Fig. 8: Result of Experimental

Fig.8 shows the curve obtained from the data. The Fig.8(a) Show the recognition time and Fig.8(b) shows the time of training 600 images with Map/Reduce model. The results show that a typical face recognize algorithm can be implemented in a distributed system like hadoop using commodity hardware and achieve acceptable execution times close to real time. We make efficient use of the computational and storage resources of Aliyun.

VII. CONCLUSIONS AND FUTURE WORK

In this paper, Cloud Robot Visual Platform is proposed. The goals of the platform are to offload the intensive workloads of robot and to shorten the image recognition time. Map/Reduce parallel computational model is adopted for the train of face library and recognition. The H264 and RTP protocol are adopted to decrease the transmission time. All modules encapsulated with service oriented architecture. Experimental results show that platform is able to cut down the recognition time of the image. There are still quite a few problems to be solved. Image recognition algorithm and compression algorithm need to be further improved. The current video transmission speed is not fast enough to satisfy the requirement and need to further improved.

ACKNOWLEDGMENT

This work is supported by National High Technology Research and Development Program of China Grant No.2009AA04Z220, National Natural Science Foundation (NNSF) of China Grant No.61375084, Natural Science Foundation of Shandong Province Grant No.ZR2015QZ08, the Fundamental Research Funds of Shandong University Grant o.2014JC034.

REFERENCES

[1] Arumugam, R., Enti, V. R., Bingbing, L., Xiaojun, W., Baskaran, K., & Kong, F. F., et al. (2010). “DAvinCi: A cloud computing framework for service robots,” IEEE International Conference on Robotics & Automation, Anchorage, Alaska, Vol.58, pp. 3084-3089, May 2010.

[2] Du, Z., Yang, W., Chen, Y., Sun, X., Wang, X., & Xu, C. “Design of a Robot Cloud Center,” International Symposium on Autonomous Decentralized Systems, 2011,pp.269-275.

[3] Tenorth, M., Perzylo, A. C., Lafrenz, R., & Beetz, M. “Representation and exchange of knowledge about actions, objects, and environments in the roboearth framework,” IEEE Transactions on Automation Science & Engineering, 2013,Vol.3, pp.643-651.

[4] Waibel, M., Beetz, M., Civera, J., Andrea, R. D., Elfring, J., & Galvez-Lopez, D., et al. “Roboearth,” IEEE Robotics & Amp Amp Automation Magazine, 2011,Vol.2, pp.69-82.

[5] Mohanarajah, G., Hunziker, D., D'Andrea, R., & Waibel, M. (2015). “Rapyuta: a cloud robotics platform,” IEEE Transactions on Automation Science & Engineering, 2015,Vol.2,pp.481-493.

[6] Gouveia, B. D., Portugal, D., Silva, D. C., & Marques, L. “Computation sharing in distributed robotic systems: a case study on slam,” IEEE Transactions on Automation Science & Engineering,2014,Vol.2, pp.410-422.

[7] Wang, L., Tao, J., Ranjan, R., Marten, H., Streit, A., & Chen, J., et al. “G-hadoop: mapreduce across distributed data centers for data-intensive computing,” Future Generation Computer Systems,2013,Vol.3, pp.739-750.

[8] Pan, Z., Healey, G. E., & Tromberg, B. (2015). “Multiband and spectral eigenfaces for face recognition in hyperspectral images,” Proc SPIE, 2015,Vol.12,pp. 144-151.

[9] Zhang, H., Liu, W., Dong, L., & Wang, Y. (2015). “Sparse eigenfaces analysis for recognition,” International Conference on Signal Processing Quebec, Candan, pp.887-890. 2015.

[10] Bahri, N., Grandpierre, T., Ayed, M. A. B., Masmoudi, N., & Akil, M. “Embedded real-time h264/avc high definition video encoder on ti’s keystone multicore dsp,” Journal of Signal Processing Systems, 2016, pp.1-18.

[11] Wu, H., Lou, L., Chen, C. C., Hirche, S., & Kühnlenz, K. “Cloud-based networked visual servo control,” IEEE Transactions on Industrial Electronics, 2013,Vol.2,pp. 554-566.

[12] Im, K. M., & Lim, J. H. (2015). “Lighting control system based on the rtp of smart grid in wsn,” International Journal of Distributed Sensor Networks, 2015,Vol.1, pp. 1-14.