A Fast Training Time Algorithm for Object Detection and Classification applied to Interactive Humanoid Robots

Daniel Tozadore
Institute of Mathematics and Computer Science
Uniservisty of São Paulo
São Carlos, São Paulo
tozadore@usp.br

Roseli Romero
Institute of Mathematics and Computer Science
Uniservisty of São Paulo
São Carlos, São Paulo
rafrance@icmc.usp.br

ABSTRACT

The presence of high interaction devices caused a decrease in the young students' attention span in traditional teaching. This fact has motivated researchers to find new ways of teaching in order to adapt the educational to the new society technological standards. Studies adopting interactive robots for pedagogical tasks have been shown a significant enhance in the students experience in such activities. Autonomous systems have also the advantage to arouse the interest in children about how this type of system works. However, the majority of techniques used to image classification have a large training time due to take in account the background noise processing and the environment changing. This paper presents a vision system integrated in a humanoid robot to play with humans, approaching a chosen subject in an autonomous way. The proposed vision system allows automatic object detection and segmentation integrated into humanoid NAO robot. It was combined vision algorithms such as VOCUS2 for object detection, SURF and BoF for features extraction and an *one-against-all* multiclass SVM for im-age classification. Tests have been performed with 3D geometric figures as cube pyramid and sphere and the obtained results show a high accuracy and a very low training time, which is desirable for quickly programmable's activities.

CCS CONCEPTS

• Human-centered computing → Laboratory experiments;

KEYWORDS

Pedagogical Robotics, Object Classification, Human-Robot Interaction

1 INTRODUCTION

Since the technology reached a large set of high interaction devices, it was noticed a decrease in the young's attention span in tradi-tional teaching ways. The strategy that have been successfully used to avoid it is to use the devices themselves to hold the students attention for a longer period. In this way, it is possible to approach the same contents in a more dynamic and enjoyable experience. These devices goes from usual smartphones and tablets to more complex systems, such as electronic circuits and robots.

Programmable robotic kits (as LEGO kits [11]) are presented as a concrete tool to support learning, leading the student to explore logical paradigms building and programming robots to perform a task related to their studies. Just like the educational games, it is an alternative way to study all the theoretical concepts, especially when the students are challenged by the technology [22]. However, with this kits we build robots like cars and animals, what makes the most of younger students to treat the robot as a toy, avoiding the robot to be used as an assistant in the learning process. To this goal, expressive and talker human-like robots can be more incisive and gain more attention of children.

Robots having the ability to interact more likely to the human behavior has become an important topic in educational robotics area worldwide. By the socially interacting with others, humans can learn more easily due to the fast and well-understandable communication that occurs naturally [21]. Following the same idea, high interactive robots can provide similar experiences that, although not totally efficient as human communication per times, has the novelty factor. Human-Robot Interaction (HRI) is a recent research area that focused in issues of this kind and it has shown a significant growth in several applications, such as health, entertainment, guidance, care-take and education. As well as that, the investigation about different types of robots and their influence in the user perceptions among tasks. The NAO robot, from Aldebaran Robotics, has a range of interaction possibilities and it is often used in this works for helping students to keep the focus in the proposed activities, aiming a higher gain in learning rates compared to appropriated control groups.

This visual system proposal is part of a bigger project named R-CASTLE project, a Robotic – Cognitive Adaptive System for Teaching and LEarning. The project aims to deliver a very innovative and advanced user-experience system as a new educational tool that allows any kind of designer (the programming and non-programming persons that want to apply this methodology) to easily plan interactive activities with electronic devices, in which in this case is applied in a humanoid robot, proving an autonomous and natural communication with the robot and adaptive skills for short and long-term interaction. Natural communication of the robot is guaranteed through modules that perform audio-visual processing and robot's gestures manipulation, while the robot's adaptation behavior relies on multiple sensors and algorithms to collect and analyze the users' perceptions. All the variables configuration from this project are result of several technical and interactive (user-centered) studies, focused in specifics issues that, combined, produce the whole scenario of this research problem. Tests with all modules integrated or tests with users were not performed in this study.

In this paper, it is only presented on of the solutions studied for the visual module, considering that the designer wants to quickly chance all the vision database to approach a different content in the following interactive sessions. The presented solution was studied due to be part of the already available laboratory However, it was important to analyze the algorithm behavior for this study goals in the submitted conditions, as presented in the following sections. Mainly, for 3D monocular object recognition it is being used Automatic Image Annotation (AIA), which is a branch of image retrieval that can be said as more user friendly. Users are much at ease if images are given with semantic keywords but manual indexing is a time consuming process. That is why the techniques under this category first annotate images with semantic keywords automatically and once images are annotated they can be retrieved much more easily.

The rest of the paper is organized as it follows. In Section 2, we discuss the background of related work using the same Automatic Image Annotation approach and Educational HRI studies. The Sec-tion 3 presents the project's proposal as well as the materials and method that compose it. The results are presented and discussed in Section 4. Finally, in Section 5, conclusions and future works are presented.

2 BACKGROUND

In turn-of-event work, Chapelle et al. [4] showed that classification can be improved based on image histograms using support vector machines (SVM). Before this, it was known that classification approaches generalized poorly on classification tasks if the dimensional of the feature space was extremely high but this approach showed that SVM can perform this classification easily if the only attributes provided are high-dimensional histograms. Chapelle et al used heavy-tailed RBF kernels. Also, they showed that decreasing a while using a-exponentiation improves performance of linear SVM that they can be used to substitute RBF kernels.

Following in this way, Goh et al. [1] used one-class, two-class and multiclass SVM. They have proposed a confidence-based dynamic ensemble (CDE) so that it can be concluded when retraining of classifier is needed and whether new low-level features or training data can be included. A three level classification scheme is proposed. At the base level, SVM are used for computing the prediction of one semantic label. A confidence factor is given for each prediction by employing algorithm for *one-class* SVM which also uses a density distribution of training data. At multiclass level, the confidence factors of all multiple classifiers are cumulated to give only one prediction. Again a multi class level confidence factor is computed for this prediction. At the bag level, CDE cumulates the predictions from multiple bags to give an aggregated prediction. An overall confidence factor is given at this level. If this is high, a semantic is assigned. This approach overcome the disadvantages of traditional static classifiers as it makes adjustments to include semantics leading to discovery of low level features and thus improving accuracy .

Shukla et al. [27] used a combination of features based techniques, as shown is Figure 1, for automatic image annotation and used the Social20 dataset [16] to test. The Social20 set has 20 visual concepts and a total of 19,972 images. Under each concept category 1000 images are present. The concepts are very sundry: airplane, beach, boat, bridge, bus, butterfly, car, cityscape, classroom, dog, flower, harbor, horse, kitchen, lion, mountain, rhino, sheep, street, and tiger. They claimed that this technique is a very efficient way to automatic image annotation, reaching 91% of accuracy and training time in 1,25s, which is a good indicator that this system can easily integrated into image retrieval system. methods.

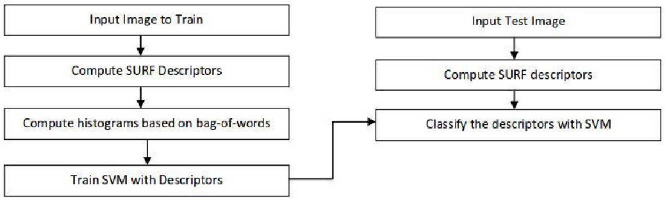

Input Image to Train
Compute SURF Descriptors
Compute histograms based on bag-of-words
Train SVM with Descriptors
Input Test Image
Compute SURF descriptors
Classify the descriptors with SVM

Figure 1: Scheme proposed by Shukla [27].

Balayil et al. [19] presented a very simple and intuitive approach to perform multilabelling of images using bag-of-words model of image representation and two pass K- Nearest Neighbor (KNN) for classification. The modified ranking approach considers the co-occurrence of labels within the dataset along with the ranking obtained using two pass KNN, hence reducing the chances of assigning those labels that hardly co-occur. Despite its simplicity, it gives reasonable performance as compared to the state-of-the art methods for solving the problem. It gives a better recall value using SIFT features while the use of SURF features makes it faster. The relation between the predicted labels are easily obtained using a predefined ontology.

Since 2012, when the AlexNet [14] won the ImageNet [6] Challenge, proving that Neural Networks with a lot of layer are now viable due to the advancing in hardware's processing power, the majority of works have been attempting to solutions using Deep Learning [15] as Convolutional Neural Networks (CNN) for image classification. As result, many architectures and models were presented and all of them with accuracy in the state-of-art but with very large training time. The more notables models the AlexNet itself, GoogLeNet and Microsof ResNet, which are the recents ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [1] winners and they are in constant improvement. The Deep Learning approach is mainly used as image recognition in many applications [26]. However, it is commonly used to other purposes as well, for example Big Data Processing [29], Natural Language Processing [28], and other Artificial Intelligence fields [2].

Visual systems are often used for human tracking in navigation robots and gestures recognition in HRI applications [23]. For interaction goals, they are majorly employed to analyze user's parameters as eye gaze [30], posture [25], emotions [17] and so on. Regarding objects identified along the interaction, few works are reported. The objects used in the interaction are specifics and it is not an advantage to train a large database due to the processing waste, once it will not serve for generic purposes.

3 PROPOSED SYSTEM

The final proposed system is a module based architecture, where each module is responsible for a specific function and contains a group of functionalities to do so. This type of implementation allows to easily change a module - or just a part of it - without impact in the other ones.

[1] http://image-net.org/challenges/LSVRC/2017/index, accessed in September 2017

For instance, if we want to use another robot it is necessary just change the modules that communicates with the robot, without interfere in the vision or dialog module. The modules are better described next and the Figure 2 illustrates how their integration works.

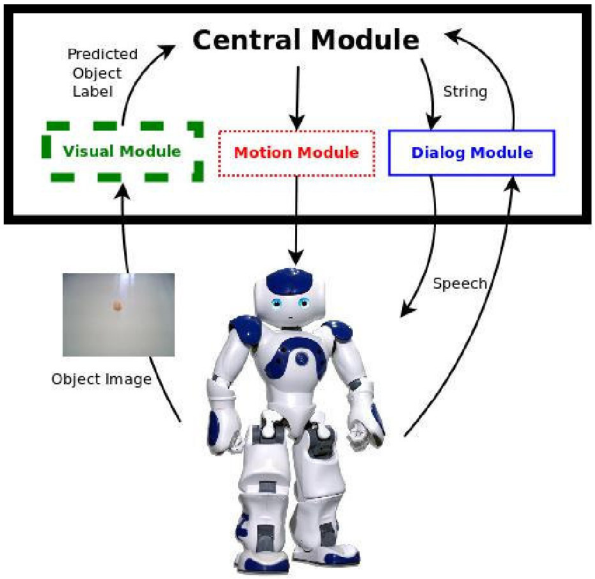


Figure 2: The architecture's modules and its communication flow.

3.1 Modules

3.1.1 Central Module. Is the main module that contains the others. Uses objects for the communication between the modules and also some useful mechanisms to help and guide the interaction flow. It connects with the robot by proxies and with the modules by function calls.

3.1.2 Dialog Module. The Dialog Module has two functions: Interpret what the user says by converting speech into text, and give information to the user by converting given sentences by the central module into robot's speech. In this way, the Google Speech Recognition is being used, which is an API that sends a wave file format to its server and takes back the corresponding string. This communication is programmed in python, with the library SpeechRecognition 3.1.3 [8], and it communicates with the module by file stream. For the voice synthesis we use NAO's default voice, as it was well accepted in other test.

3.1.3 Motion Module. In the Motion Module are implemented with NAOqi SDK functions to position the robot's engines. Is always used with another module, but very import to hold users attention span for a longer period.

3.1.4 Vision Module. More relevant to this paper, the Visual Module is composed by a series of techniques to detect and recognize the chosen objects. First, it is employed VOCUS2 for segmentation and background and noises extraction. Then, computed the SURF features and bag-of-words method in their histogram, and, finally, multiple SVM - which is a supervised method - are trained with this descriptors to annotate, or predict, new input images. All of these employed methods for vision are better described in Subsection 3.3.

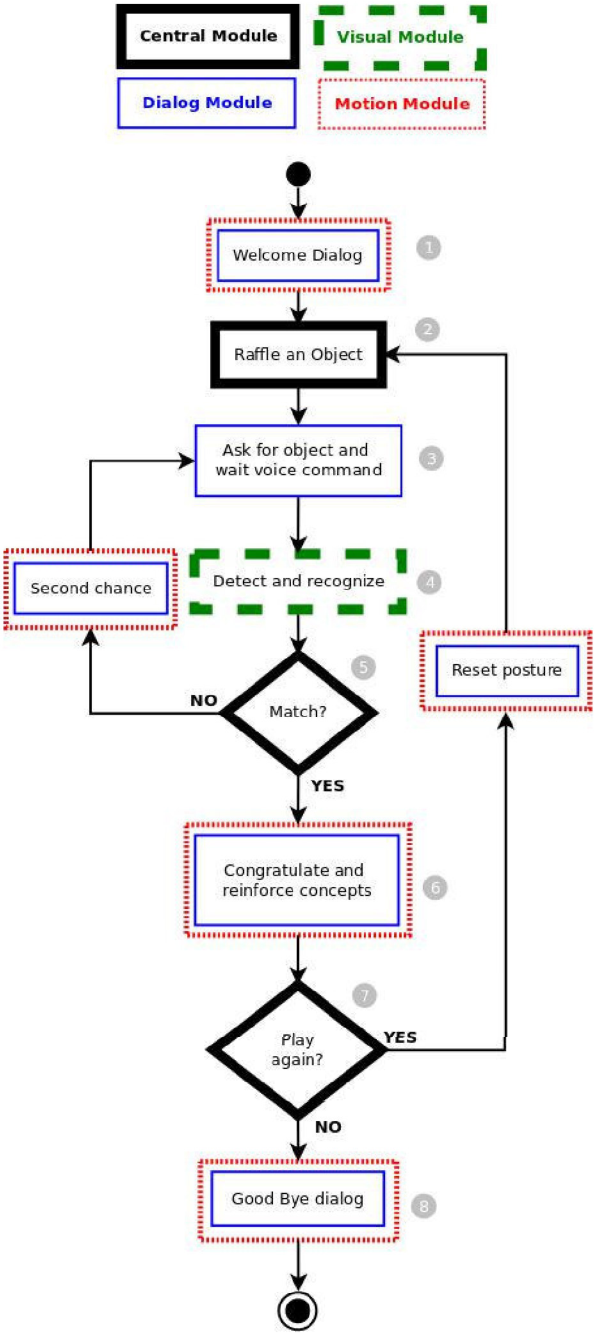


Figure 3: Session's Flow Chart. The borders corresponding to the module operating in that step.

3.2 Interaction Design

The system was configured to play with young students in interactive sessions asking and recognizing three types of basic 3D geometric figures - sphere, cube and pyramid - and explaining their differences and concepts. These figures were made from plastic by a 3D printer, except the sphere that is a simple table tennis ball. The reason of the chosen content is that, preliminary studies performed shown a big mistake by children to differ 2D and 3D geometric figures, and also due to its simplicity, since these are the first tests. Thus, for pedagogical matters, the dialog module was set with sentences addressing the concept of vertex, edges, faces, bases and things that characterize these adopted objects. It is worth to remember that, replacing the content from dialog and visual modules is possible to deal with so many subjects as wanted.

The set of interaction between the robot and user is called a session and the combination of techniques described are used to guide the robot along the interaction flows. Considering the selected subject and with help from math education specialists, it was suggested the session presented by the flow chart in Figure 3, in which the interaction flow is composed by 8 steps from the main flow and 2 complementary steps due to the possible changes in some steps. To better illustrates, all the steps in figure have borders according their modules and some steps are performed by more than one module.

In the step 1, the robot performs a little welcome dialog with the children, having a hand-shake and asking the children's name to use along the session. In the step 2, the central module randomly chose one of the three possible objects. In the step 3, the dialog module requests the chosen object, but without say its name, just giving tips and features. An example for a cube could be: "Could you please put in front of me a 3D geometric figure with 8 vertexes and 12 edges?". In the step 4, the visual system predicts the object and returns the prediction to the central module, that in the step 5 checks if the object chosen by the children matches with the choice in step 2. In negative case the robot gives more tips and regress to the step 2, requesting for the figure again. In positive case the flows proceed through the step 6, where the robot congratulates for the right choice and gives some complementary concepts or curiosities about the object. In step 7, the central module can be set to play a determined number of rounds or send a message, through the dialog system, if the children want to play again. If there is another round, the steps are reproduced from step 2. In other case, there is a goodbye dialog in step 8 and the system ends its work. An example of children interacting with NAO is shown in Figure 4.

3.3 Vision Module

The following subsections detail the methods used in the Vision Module.

3.3.1 VOCUS2. The VOCUS2 [10] is a salience system that follows in its basics structure the Itti-Koch [12] model: feature channels are computed in parallel, pyramids enable a multi-scale compu-tation, contrasts are computed by Difference-of-Gaussians (DoG). The most important difference is the scale-space structure (that use a new twin pyramid) and the center-surround ratio, which has turned out to be the most crucial parameter of saliency systems.

Figure 4: Children interacting with NAO during the pro-posed session.

Similar to the approach of Itti, the resulting system has a simple and elegant structure which follows concepts from human perception, but this system produces pixel-precise saliency maps, instead of segment-based ones. However, the segment-based saliency maps are beneficial for some tasks, especially if precise object boundaries are required. VOCUS2 system works as shown in Figure 5. The input image is converted into an opponent-color space with channels for intensity, red-green, and blue-yellow colors. For each channel, it computes two image pyramids (one center and one sur-round pyramid), on which center-surround contrast is computed. This structure is similar to others FIT-based such as iNVT [12] or VOCUS [9].

3.3.2 SURF Feature Descriptor. As features in an image can be found very easily now due to distinctive use of descriptors that can be computed on the whole image rather than its segmented parts, it has become easier to be more accurate. SIFT, PCA-SIFT, SURF are some such descriptors. One of the very reasons for their popularity is that they are invariant to image rotation, scaling, changes in illumination. The reason SURF is preferred over SIFT is due to its concise descriptor length. Whereas the customary SIFT implementation uses a descriptor consisting of 128 floating point values, SURF compresses this descriptor length to 64 floating point values.

3.3.3 Bag Of Features. Bag of Features (BoF) is a popular approach to visual object classification whose interest is due to its power and simplicity. Its origin stems from the bag-of-words model [24]. This approach is used for many computer vision tasks, such as image classification, robot localization and textures recognition. The methods that apply this model are based on unordered collections of image descriptors. They have the characteristic of discarding spatial information and are conceptually simpler than alternative methods.

The idea, as shown in Figure 6 in this approach is to compute features in an image and match with a set of features to classify the image. A feature is a property that can represent an image or part of it.

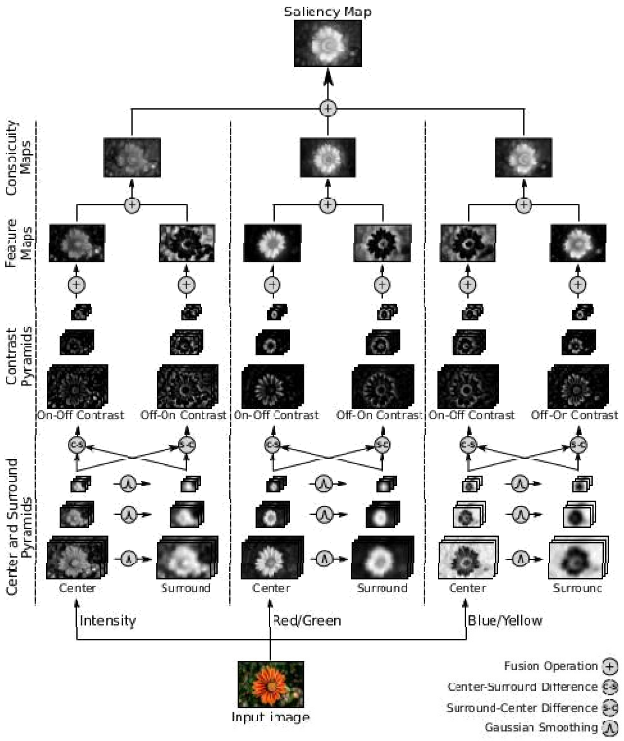


Figure 5: Basics of VOCUS2 system [10].

It may be a pixel, a circle, a line, a region with texture, medium gray level, etc. Despite that there is no formal definition, features can be characterized as detectable parts of the image with some meaning. In this model, the extracted features are grouped and generated partitions are used to mount a dictionary of visual words. After quantify the features using the visual dictionary, the images are represented by the frequency of visual words.

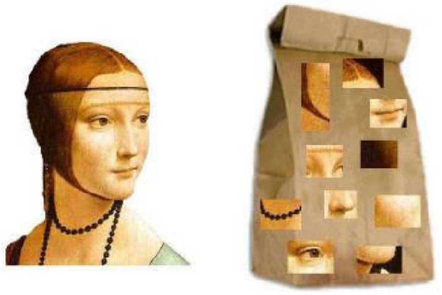


Figure 6: Bag of Features [7]. The image of a face and a bag formed by feateures that caracterizes a face.

3.3.4 Support Vector Machine (SVM). The Support Vector Machines (SVM) deals with finding a predictive function to generalize data. SVM is used in several researches, as Pattern Recognition [5], Object Recognition [3], speaker identification, face detection, text categorization, for example. In most of tests, compared to bench-marks and other classic approach, SVM generalization performance (error on test sets) matches or is significantly better.

The difference between the SVM and other classical machine learning algorithms is the bias assumed. As ID3 or MLP, the bias for the SVM is based on linear functions. However, are linear on the features space, and can provide a non-linear classifier. Data can be not separable in a given input space, so it can be mapped to another space and another dimension with similarity measure. This mapping function is called Kernel, and it is important to understand your input space and the distribution of your data to find the best kernel that will classify with less error.

In this paper, linear SVMs classifiers are used using the above provided inputs. Once the bag-of-words features for all training images are obtained, they are given into SVMs. They find a hyper plane that separates the training data by maximal margin. *"One-against-all"* approach is used in the framework as it achieves comparable perfor-mance with faster speed than *"one-against-one"*. In the *one-against-all* implementation of SVM, *n* hyper-planes are implemented, where *n* is the number of classes. Each hyper plane can be used to separate one class from the other classes.

4 VALIDATION TESTS

In order to obtain better measures regarding the efficiency in correctly classify the objects, the visual module was submitted to a series of tests. In detection, the VOCUS2 was successful in detect all the test samples, demonstrating that it fits too well this application, removing background and isolating the object. As described in Section 3.1.4, the segmentation is the first step in detecting and processing the image provided by the robot's camera. The Figure 7 shows an example of the segmentation process.

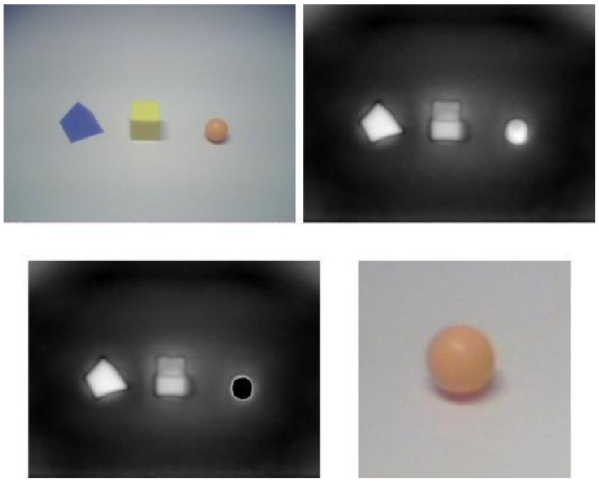
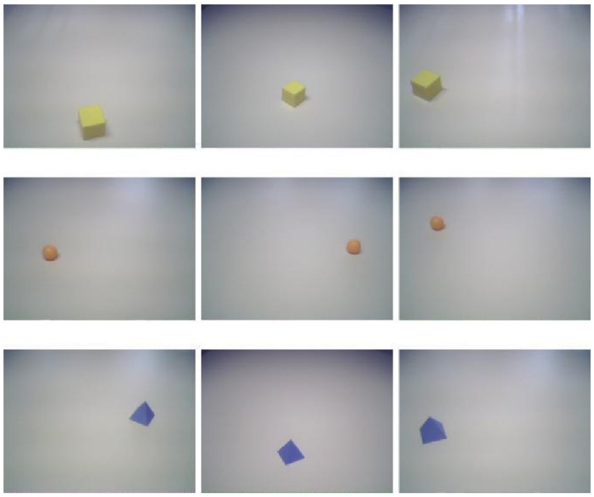


Figure 7: Object detection process: 7(a) 400x400 pixels original image captured by the robot's camera, 7(b) the saliency map, 7(c) the most salient object on scene and 7(d) the cropped object.

Easy to note that, the original images could also serve as good entrances for training and classification, but as the final system aims to works in noisy environments and recognize more complex objects, we decided to run the complete system even with these simple objects to study its behavior.

Three SVM were train, one for each class of 3D geometric figures. The training dataset was composed by 60 samples, being 20 of each class. It is known that it is considered a very small sample set, but the focus is to investigate how the classifier reacts to few images as training inputs. The images were taken from robot's camera and their size may vary according their position in the robot's field view, it means, the distance between 3D figures and the camera. In Figure 8 is possible to see some samples from the training dataset.



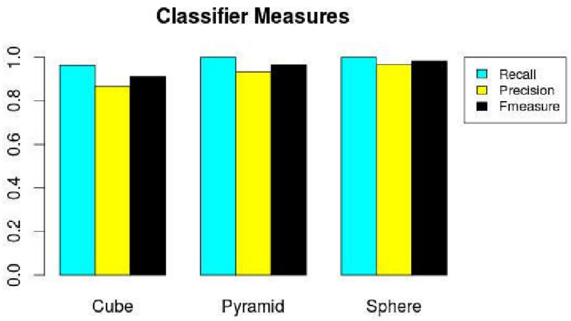


Figure 8: Samples of training dataset of each class: cube in the first row, sphere in the second and pyramid in the third.

For a performance test of the visual classification technique in a known database, please check [27] and for a vehicles database annotation check [18]. In this study the system was tested with a testing dataset composed by 90 samples distinct from the training dataset, being 30 images of each class and varying their positions. In Table 1 it is shown the confusion matrix for the test dataset automatic annotation. The lines are the class that the sample belongs and the columns are the SVM predictions. Finally, the Unknown class in the last column indicates that none of SVM were able to predict that sample, it means, that sample was not recognized in any class by the multiple SVM.

Three measures were used to evaluate the classification acuracy: Recall, Precision and F-Measure. The recall is the number of correct predictions divided by the number of occurrences of keyword with ground truths in the test dataset. In other words, the recall of any classifier is computed as dividing the correctly classified positives by total positive count of images that are been tested [20]. The precision is the number of correct annotations divided by no of predicted annotations. In other words, it is number of correctly retrieved images divided by the number of retrieved images [13]. To combine recall and precision in a single efficiency measure, the harmonic mean of precision and recall is calculated. It is called F-measure (Equation 1). This is one of the aggregated performance measures. The results for the three measures in the presented tests are shown in Table 2 and in the graph of Figure 9.

Three measures were used to evaluate the classification acuracy: Recall, Precision and F-Measure. The recall is the number of correct predictions divided by the number of occurrences of keyword with ground truths in the test dataset. In other words, the recall of any classifier is computed as dividing the correctly classified positives by total positive count of images that are been tested [20]. The precision is the number of correct annotations divided by no of predicted annotations. In other words, it is number of correctly retrieved images divided by the number of retrieved images [13]. To combine recall and precision in a single efficiency measure, the harmonic mean of precision and recall is calculated. It is called F-measure (Equation 1). This is one of the aggregated performance measures. The results for the three measures in the presented tests are shown in Table 2 and in the graph of Figure 9.

$$F\_measure = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \quad (1)$$

Table 2: Measures

	Recall	Precision	F-Measure
Cube	0.96	0.86	0.91
Pyramid	1	0.93	0.96
Sphere	1	0.96	0.98

Table 1: Confusion Matrix.

	Cube	Pyramid	Sphere	Unknown
Cube	26	0	0	4
Pyramid	0	28	0	2
Sphere	1	0	29	0

Half of the figures were positioned in the middle range of the robot's view field and the other half scattered in the periphery. All the objects in the middle distance were correctly predicted, while the prediction mistakes were made for the objects farthest from the center of the Robot's view field .

This implementation showed to be adequate, since it was obtained 93% of accuracy and training time in 0.9s for the chosen 3D geometric figures as cube, pyramid and sphere. The performance in all the measures presented by the proposed approach combined with the very low training time suggest that the this trade-off fits well the architectures goals. However, further investigation is required to analyze the visual system in other conditions.

Other techniques, as Convolutional Neural Networks, could also be used to classify the images with higher accuracy and other advantages. However, they present larger training time, which is out of the final project scope regarding providing faster programmable activities to the students' professors. In technical matters, studies comparing this vision system with other methods to compare parameters as accuracy, training time and other contribution to the final application are being conducted so far. While for application validation, studies with the whole architecture are also being performed with several young students.

REFERENCES

[1] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. 2006. Surf: Speeded up robust features. In Computer vision–ECCV 2006. Springer, 404–417.

[2] Yoshua Bengio et al. 2009. Learning deep architectures for AI. Foundations and trends® in Machine Learning 2, 1 (2009), 1–127.

[3] V. Blanz, B. Scholkopf, H. Bulthoff, C. Burges, V. Vapnik, and T. Vetter. 1996. Comparison of view–based object recognition algorithms using realistic 3d models. In C. von der Malsburg, W. von Seelen, J. C. Vorbruggen, and B. Sendhoff, editors, Artificial Neural Networks — ICANN, pages 251 - 256, Berlin, 1996. Springer Lecture Notes in Computer Science, Vol. 1112.. (1996).

[4] Olivier Chapelle, Patrick Haffner, and Vladimir N Vapnik. 1999. Support vector machines for histogram-based image classification. Neural Networks, IEEE Transactions on 10, 5 (1999), 1055–1064.

[5] S. Cortes and V. Vapnik. 1995. Support Vector Machines. Machine Learning, 20:273–297. (1995).

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 248–255.

[7] Li Fei-Fei, Rob Fergus, and Antonio Torralba. 2005. Recognizing and Learning Object Categories. Short Course at ICCV 2005. (2005).

[8] Python Software Foundation. 2014. Available in: https://pypi.python.org/pypi/SpeechRecognition/. Last seen at 02/20/2107. (2014).

[9] Simone Frintrop. 2006. VOCUS: A visual attention system for object detection and goal-directed search. Vol. 3899. Springer.

[10]Simone Frintrop, Thomas Werner, and Germán Martín García. 2015. Traditional Saliency Reloaded: A Good Old Model in New Shape. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 82–90.

[11] Shakir Hussain, Jörgen Lindh, and Ghazi Shukur. 2006. The effect of LEGO training on pupils' school performance in mathematics, problem solving ability and attitude: Swedish data. Journal of Educational Technology & Society 9, 3 (2006), 182–194.

[12] Laurent Itti, Christof Koch, and Ernst Niebur. 1998. A model of saliency-based visual attention for rapid scene analysis. IEEE Transactions on Pattern Analysis & Machine Intelligence (1998), 1254–1259.

[13] Jiwoon Jeon, Victor Lavrenko, and Raghavan Manmatha. 2003. Automatic image annotation and retrieval using cross-media relevance models. In Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. ACM, 119–126.

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems. 1097–1105.

[15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. Nature 521, 7553 (2015), 436–444.

[16] Xirong Li, Cees GM Snoek, and Marcel Worring. 2009. Learning social tag relevance by neighbor voting. Multimedia, IEEE Transactions on 11, 7 (2009), 1310–1322.

[17] Daniele Mazzei, Abolfazl Zaraki, Nicole Lazzeri, and Danilo De Rossi. 2014. Recog-nition and expression of emotions by a symbiotic android head. In Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on. IEEE, 134– 139.

[18] R. Montanari, E. C. Fraccaroli, D. C. Tozadore, and R. A. F. Romero. 2015. Ground vehicle detection and classification by an unmanned aerial vehicle. LARS/SBR 2015, 2015, Uberlândia - MG. Proceedings of LARS/SBR 2015, 2015. p. 253-257.

[19] G. Santhosh Kumar Munia Balayil and V. Muhammed Anees. 2014. Automatic Multilabelling of Images and Semantic Relation Extraction. Journal Issue on Imagin and Signal Processing (2014).

[20] David L Olson and Dursun Delen. 2008. Advanced data mining techniques. Springer Science & Business Media.

[21] Dennis Perzanowski, Alan C Schultz, William Adams, Elaine Marsh, and Magda Bugajska. 2001. Building a multimodal human-robot interface. IEEE intelligent systems 16, 1 (2001), 16–21.

[22] A. H. M. Pinto, A. X. Benicasa, L. O. Oliveira, R. C. G. Meneghetti, and R. A. F. Romero. 2014. Attention Based Object Recogniton applied to a Humanoid Robot. LARS/SBR 2014, 2014, São Carlos. Proceedings of LARS/SBR 2014, 2014. p. 136-141. (2014).

[23] Siddharth S Rautaray and Anupam Agrawal. 2015. Vision based hand gesture recognition for human computer interaction: a survey. Artificial Intelligence Review 43, 1 (2015), 1–54.

[24] Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. Information processing & management 24, 5 (1988), 513–523.

[25] Jyotirmay Sanghvi, Ginevra Castellano, Iolanda Leite, André Pereira, Peter W McOwan, and Ana Paiva. 2011. Automatic analysis of affective postures and body motion to detect engagement with a game companion. In Human-Robot Interaction (HRI), 2011 6th ACM/IEEE International Conference on. IEEE, 305–311.

[26] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. 2014. CNN features off-the-shelf: an astounding baseline for recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops. 806–813.

[27]Tuhin Shukla, Nishchol Mishra, and Sanjeev Sharma. 2013. Automatic image annotation using SURF features.

International Journal of Computer Applications 68, 4 (2013), 17–24.

[28] Richard Socher, Yoshua Bengio, and Christopher D Manning. 2012. Deep learning for NLP (without magic). In Tutorial Abstracts of ACL 2012. Association for Computational Linguistics, 5–5.

[29] Ian H Witten, Eibe Frank, Mark A Hall, and Christopher J Pal. 2016. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann.

[30] Abolfazl Zaraki, Daniele Mazzei, Manuel Giuliani, and Danilo De Rossi. 2014. Designing and evaluating a social gaze-control system for a humanoid robot. IEEE Transactions on Human-Machine Systems 44, 2 (2014), 157–168.