The use of computer vision technologies to augment human monitoring of secure computing facilities

Marius Potgieter
School of Information and
Communication Technology
Nelson Mandela Metropolitan University
Port Elizabeth, South Africa 041 504 1111
Email: s208108589@live.nmmu.ac.za

Dr Johan Van Niekerk
School of Information and
Communication Technology
Nelson Mandela Metropolitan University
Port Elizabeth, South Africa 041 504 3048
Email: johanvn@nmmu.ac.za

***Abstract*—Humans are poorly equipped to perform repetitive tasks without adversely affecting the efficiency with which they are performing the task. Assets within a secure environment are usually protected with various controls that are enforced by users who follow operational controls associated to those assets. The current approach to security monitoring by means of video cameras are performed by a person physically needing to concentrate on multiple video feeds. This method relies on the constant vigilance of the operator and the consequence of loss of vigilance can range from minor theft of assets to missing a bomb placed within an airport.**

This paper will approach security monitoring using Computer Vision augmented with Speeded-Up Robust Features (SURF) as the catalyst to provide event-driven object detection to assist in securing an environment. A scenario of a secure computer environment will be used to demonstrate the problems with current approaches and present an alternative to human monitoring using Computer Vision. The paper demonstrates that some of the physical aspects of information security can be improved through the use of SURF algorithms.

I. Introduction

The modern world is a fast paced environment where access to information and the ability to respond to constant change is vital to organizational success. Nowadays, access to information is so crucial that some authors no longer see it as a competitive advantage, but rather as a must have commodity, similar to electricity [1]. It is thus vital to protect organizational information against harm or loss. The process of protecting information is known as information security.

Information security is commonly implemented in the form of various information security controls. These controls are described in International Standards such as ISO/IEC 27002 [2] or ISO/IEC 13335 [3]. These standards provide three broad categories of Information Security controls; namely physical, technical and operational controls.

An example of a physical control is a physical lock on the door to a computing facility. An example of a technical control is the requirement for users to be identified and authenticated through some form of log-on procedure before allowing them access to the organizations information resources. Operational controls include all controls dealing with the role(s) of humans in the security process. Both physical and technical controls are often dependent on supporting operational controls in order to be effective. An operational control might, for example, thus state that the physical door to the computing facility must remain locked and that all users must log-off when leaving their workstations. Failure to adhere to operational controls can thus negate the usefulness of the other types of controls.

It is thus vital that compliance to procedures outlined by operational controls are established within the organization. Most current research recommends awareness, training and education and/or the establishment of an information security culture to improve such compliance. However, it would be beneficial if one could enforce compliance to these information security controls, instead of only relying on voluntary compliance. This paper will specifically examine the enforcement of policies governing all aspects of the physical control of information assets through the use of computer vision technologies to support the monitoring of a secure computer facility.

The remainder of the paper will introduce a scenario where the physical monitoring of a secure computer environment is important. The paper will then introduce Speeded-Up Robust Features (SURF) algorithms and demonstrate how such techniques can be used to augment current camera-based monitoring of such secure environments in order to reduce reliance on human operators.

II. Scenario

A. Introduction

The protection of mission critical computing facilities often take the form of a computer room with various technical controls to provide security. The controls typically includes access control and physical monitoring performed by video cameras through a CCTV system.

Figure 1 shows such an example computer facility containing equipment that needs to be monitored. Both access to and possession of equipment is generally controlled.

In this facility the access control systems provide the technical control of the physical locking system that allows entry to the secure environment. The supporting operational control would be to allow only certain people access to this area.

The video cameras provide physical monitoring of the area using video cameras usually on a CCTV system. The video

978-1-4673-2159-4/12/$31.00 ©2012 IEEE

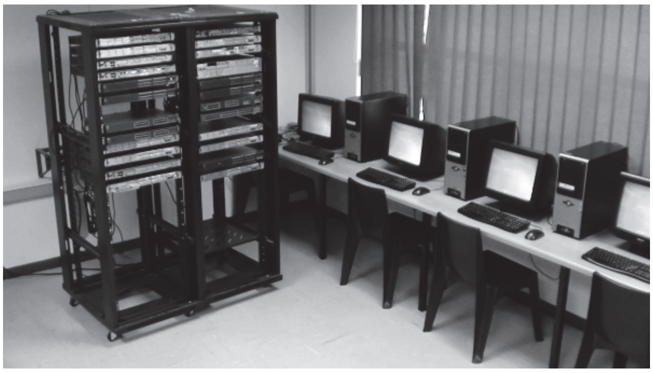

Fig. 1: Computer Facility

streams can also be recorded and stored for archival purposes. To enforce operational controls based on the footage a person needs to monitor it.

B. Problems

One problem with many security controls are that they do not provide foolproof solutions to the outcomes required by operational controls. In the case of access control systems you would have people granting access to the secure environment to someone that would not normally have access. Granting access to unauthorized people exposes the secure environment to a variety of threats.

Similarly video monitoring needs to be performed by a human to act on threats to the environment. Depending on the amount of footage to monitor this can create unrealistic assignment of human resources to such a task. Even this does not provide foolproof monitoring since threats to the environment is the exception and not the norm. Research have shown that the person monitoring can get complacent and not be alert for possible threats due to a limited capacity to maintain attention span during repetitive tasks [4]. According to Harris humans are poorly equipped to detect low-sign-to-noise-ratio signals embedded in the context of varying background configurations.

These problems only outline some vulnerabilities that controls need to deal with. Global surveys found that insider threats pose the greatest danger to controls placed to secure assets [5]. This provides a greater risk to security from an insider than that from external sources. Shifting the responsibility of monitoring from a human to that of a computer based monitoring system can firstly reduce human error induced by factors such as fatigue and complacency. Computer based monitoring will also reduce the risk posed by insider threats. Such computer based solutions does not necessarily have to replace humans, but can augment human monitoring in order to reduce fatigue and improve monitoring efficiency.

The use of algorithms to assist with computer monitoring of video data is often hampered due to imprecise data. This includes scale and perspective variations that are caused by viewing an object from difference distances and angles [6]. This paper will suggest using Speeded-Up Robust Features (SURF) as a tool to augment computer vision to allow for more robust object tracking and monitoring.

III. Implementation

A. Introduction

Object recognition has been a focus point in machine vision research for the past decade for the advancement of robotics, security, defense technology and other related fields. The processes mentioned in the different research materials will be examined to provide insight into different aspects of object recognition that will ultimately form the basis on which the holistic approach to a generic object recognition engine that will facilitate a secure monitoring system.

The process of scene modeling and recognition has been thoroughly explored and defined where interest points are detected, descriptors are generated and features generated with those descriptors. These features describe an image and thus the objects within this image. The algorithm that has been used with great success in computer vision is called Scale-invariant feature transform (or SIFT). It should be noted that most algorithms within computer vision including SIFT are performed on grey-scale images.

The following steps is followed to apply a SIFT algorithm on an image to support Computer Vision:

1) Image Processing Collect image from source and convert to grey-scale
2) Image Analysis Interest Point generated on source image and features generated around them
3) Pattern Recognition Compare an match interest points/features from source with objects within security database being monitored
4) Event Detection Attach events to objects that have been interacted with in source

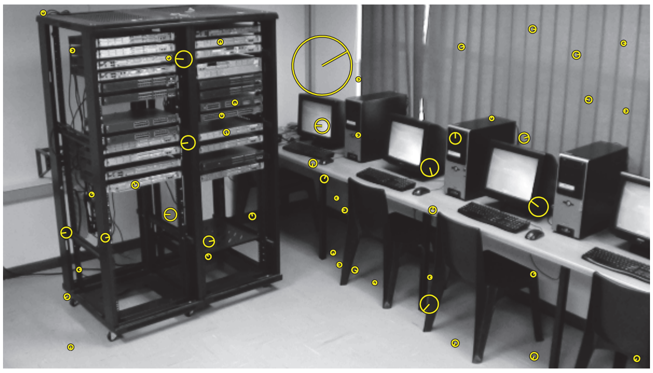

Fig. 2: Image with a few interest points with their scale and rotation angle drawn

This variant of feature descriptors also describes an image at set interest points. The interest points are selected using algorithms that define areas within an image that looks most distinct and can be well defined as shown in Figure 2. Around these interest points the feature algorithm calculates the maxima and minima in the difference of Gaussians function. A difference of Gaussians function discards details of an image as the image gets blurred at different levels (also referred to as scale-space) and compare with the original image. The change that occurs between images will be the Gaussian Distance.

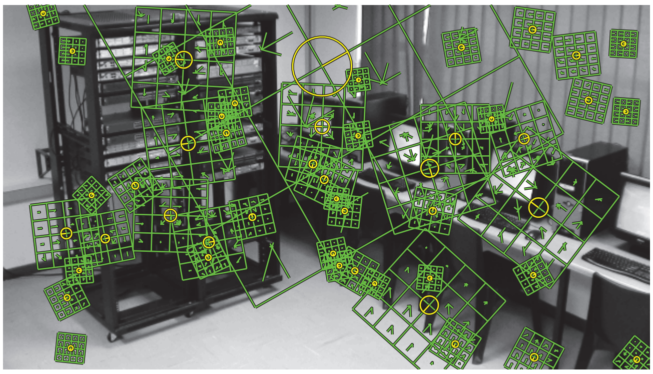

Fig. 3: Interest points with their corresponding histograms

By obtaining the maxima and minima of this function we obtain a feature that consists of vectors that define the gradient intensity and general direction around those points. Each individual vector can then form part of a subsection of the feature that can be defined as a histogram as shown in Figure 3. Each subsection can be summed into a general direction and scale which provides a summary of the feature.

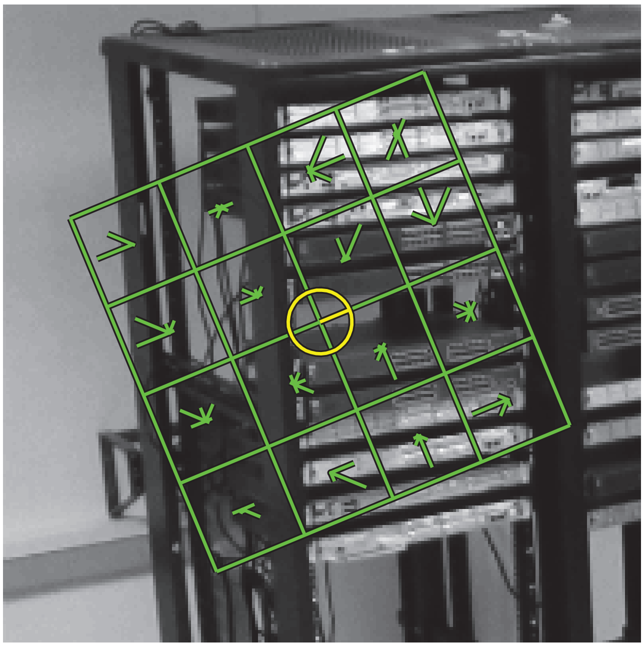

Fig. 4: Single interest point and a the histograms defining the summed area around it defined by vectors

This histogram, which contains our distribution of vectors, defines our feature as a list of integers that corresponds to the histogram values. It is this list we use to compare to other images, matching features to obtain a positive match (Figure 6). The matching process uses nearest neighbor and Best Bin First search techniques to increase the speed of indexing and matching.

The variant of features we will suggest to be used with our computer vision technology is called Speeded-Up Robust Features [7]. They are basically calculated the same way as SIFT features but have been improved to increase performance when performing matches. The way it achieves this is by using integral images, areas that have been summed within a grid of values. The comparison can be seen with SIFT with the various histogram values that form the feature but in the case of SURF these subsections are summed to form so-called Haar-like features. Haar-like features are similar to Haar wavelets that are square shaped functions that form predefined variations of the feature. The simplicity of the features makes it easy to evaluate and thus gives it a speed advantage compared to the more sophisticated SIFT variation. The speed at which the computer vision algorithm performs the matching is important since our goal is to perform real-time detection of events within an environment containing multiple objects that needs to be detected and matched with the database of objects.

The use of categorization before classification in object recognition is suggested to decrease the pool size of images that is used to compare descriptors. The use of matching kernel functions to compare descriptor vectors significantly reduces processing required to describe an image by using this kernel in a support vector machine (SVM) [8]. A SVM is a concept in statistics and computer science for a set of related supervised learning methods that analyse data and recognize patterns.

The settings for determining the threshold of these images is traditionally manually set depending on its use. This does not favour a generic system that could cater for a variety of image conditions. Investigation into using genetic algorithms with image processing has determined that an adaptive learning system can be used to adjust these parameters using genetic algorithms with a neural network [9].

Image segmentation is an important part of object recognition which defines the process of separating individual objects from an image containing multiple objects. This allows for feature recognition to occur on an object-to-object basis without having to account for the background or surrounding objects [10].

The above mentioned related research will be incorporated in one form or another to create the Computer Vision Engine. The resulting engine will incorporate features that provide the performance enhancements required to process high quality images as well as create a generic engine capable of processing any range of image variations. Additional functionality such as learning algorithms (genetic algorithms and neural networks) as well as incorporating image segmentation and categorization that focuses on an adaptive approach to object recognition was mentioned. This will not be the focus of this paper but will be the topic of future research for use in security monitoring.

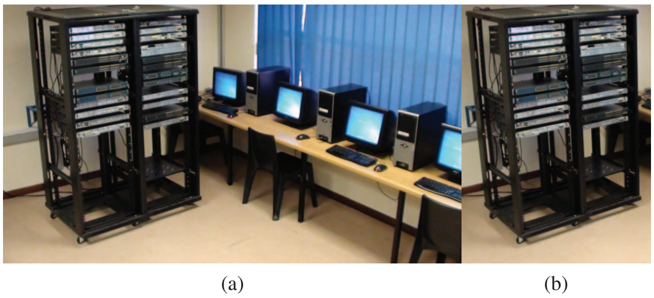
(a) (b)

Fig. 5: (a) Contains frame as seen from video source. (b) Image stored in database and categorised as the object it represents and its ownership. This image will have its features compared to that of the frame in the video.

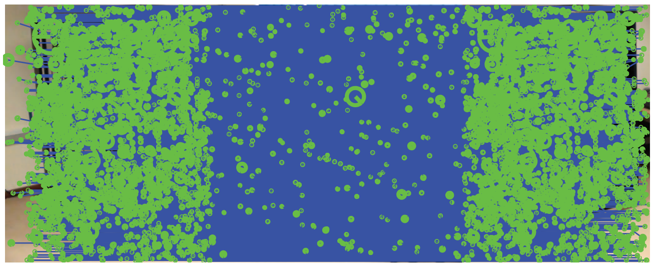

Fig. 6: Interest points are matched between the video frame and this one object in the database containing other objects. As can be seen here there is clearly a high match for the object.

B. Computer Vision-Technology

Feature descriptor algorithms generally consist of interest point detection followed by descriptor calculation. The proposed algorithm is to include steps between the previously defined steps to increase its performance and accuracy. To simplify the process these steps have been classified into phases.

The following phases represent the changes to the process:

1) Image processing, deals with the preparation of the image for feature extraction which will include creating separate threads for processing each image in smaller parts;
2) Image analysis, where the interest points will be detected which will provide logical objects to focus the detection on;
3) Pattern recognition, matches the detected feature descriptors to the predefined library of objects through classification then recognition; and
4) Event Detection will provide information about the detection object either as a still image or video in relation to its position, relative distance to other objects and other parameters which enable an event driven object engine.

1) Image processing: This phase of the process involves acquiring the image from the camera. The ability to locate objects in a two-dimensional space image is one of the main obstacles of object recognition. The use of canny edge detection is used to distinguish between various objects within a static image. The Canny Edge Detection algorithm takes the derivative of an image to find the gradients, this determines the directional gradients. The amplitude of the gradients is used to either include or exclude the gradient as part of the contours forming the edge. Once edges of objects are found pass these objects through the image analysis phase that will implement the SURF algorithms.

Acquire Video Feed as Images:-

This step simply acquires sets of images from the video source. Video cameras possess different levels of frame rates rated in Frames per Second (fps). Since most cameras work between 20-30 fps we will assume that within one second there will not be much difference in the frames caught in that second to justify processing each frame. Thus only one single frame is taken from each acquisition period that will be set manually by the discretion of the user.

Image Resampling and Combination:-

Since we are trying to apply a generic approach to video monitoring we will be resampling the image into standard resolutions and quality to facilitate the analysis phase. Image transformation that increases the efficiency of key point matching will be applied using different image filters [11]. The use of genetic algorithms to adjust the parameters for the image filters has been suggested when dealing with image segmentation and could supply a viable alternative to manually setting parameters of other types of filters [10]. Each captured frame will be packaged with all its associated images that have been resampled (in this case to grey-scale versions of the frames).

2) Image analysis: The SURF algorithm is used to form the integral image (summed area table) which uses subsets of the image in grids and sums them [12]. This increases the efficiency of Hessian Blob Detectors that detects points or regions that differs from their surroundings. Using the box filters we locate interest points that would best fit that given box filter in a specific Hessian determinant Figure 4. Calculating these box filter responses is extremely expensive on processing power, requiring 126 million lookups for a 1280 x 1024 image. It has been suggested that one can decrease the bottleneck in this multi-pass approach by sharing the results across each pass [13]. We can improve this by creating a parallel computing environment where the power of current processors are used to run all passes at once and select the best result from the output. Once the image have been reduced to an integral image, consisting of interest points, the resulting output can be converted into Haar Filters that will form part of each individual SURF-descriptor.

These SURF-descriptors collectively will form the images pattern recognition signature that will be used to compare with trained objects in the database.

3) Pattern recognition: Pattern recognition is performed by comparing the set of SURF-descriptors with those of the trained images. When a match occurs that is at an acceptable tolerance level the match gets accepted as the object being observed. This method however gets exponentially more demanding on processing power depending on the size of the trained image database. The use of a SVM (Support Vector Machine) to classify local descriptors in order to perform object categorization can significantly reduce the amount of objects the process needs to compare [8]. The kernel function that categorizes the input descriptors creates a structure to identify an object without referring to the individual descriptors. This allows the use of an index system to narrow the parameters of the set used for comparison. If the observed object is a close match to one of the indexed categories, the collection assigned to it will be loaded into memory and each individual set object will be compared to make a more accurate recognition. The result of the comparison should have more matches to an object within the loaded set, if this is not the case the object should be trained as an object that has not yet been recorded. Additional information about trained objects can be added at this stage that could be used to assist additional classification. This could also assist in creating a vocabulary tree that would provide scalability of the database [14]. Potential automatic classification of an object can be implemented through the use of online image galleries such as Google Images [15].

4) Event Detection: The matched object contains a series of matching features which can be plotted on a homography matrix. The homography matrix provides detail about the transformation between the observed image and the object in the database that has been trained. By examining the homography matrix the mapped transformations can be used to determine the object's general rotation, location and alignment to its trained version as well as other objects that have been matched. The state of these conditions can be used to provide event based triggers that can be used to create event handlers. These handlers assist in creating interactive object recognition systems as well as reducing the continuous recognition of the same object by tracking its current position and excluding it from the pattern recognition phase.

It is at this point where operational controls can be applied to objects that have been observed in order to provide security of those objects. In the case of our scenario of a secure computer facility we will have an object database of computer equipment and some generic human models to track. Objects and the humans that interact with them will provide events that we can use to alert users of possible security threats. This automated process provides security monitoring an alternative to current approaches where a user was needed to be on the lookout for all these events on multiple video sources. The result of this suggested system is to only alert users of the events detected by the system that was flagged by the system as a possible security threat. The flagging process is not discussed within this paper but forms part of algorithms matching events detected in the computer vision system to security operational controls.

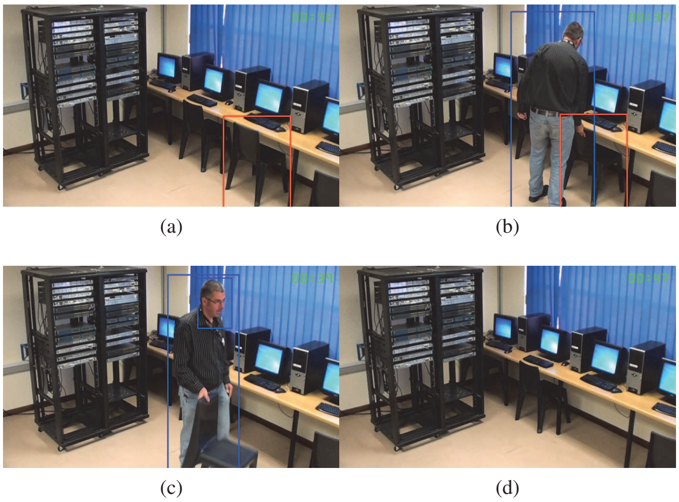

Fig. 7: Four frames from a video taken at certain time sequences illustrating event tracking.

IV. Discussion

This research was based on the premise that SURF requires less processing power than SIFT which means that SURF should be better at processing real-time footage. The previous section demonstrated how SURF algorithms could be used to perform such image processing and why this would be ideal for implementing a security monitoring system. During the course of the research SURF technology was used in a program to analyze physical events in a simplistic secure computer facility. This prototype implementation demonstrated that the use of SURF technologies for this type of system is viable. The system was used to successfully detect various events in the environment, such as the one shown in Figure 7. After an initial testing it was found that the SURF algorithms would need additional refinement as the amount of ”clutter” in the environment increases. Genetic algorithms has been used to make the SURF algorithms more adaptable. However a discussion of these genetic algorithms falls outside the scope of the current paper and will be presented in future work.

The prototype demonstrated that it was viable to detect actual events using realtime footage. The events still needs to be interpreted to determine whether or not they are real security incidents or normal behavior. Such interpretation has not been automated yet. However, it should be possible to do such interpretation using adaptive AI technologies like Genetic Algorithms and Neural Networks. This will form the basis of future research.

V. Conclusion

We have presented an implementation of a computer vision technology to augment human monitoring of secure computing facilities through the use of SURF. It was established that current systems that rely on users to classify possible threats over an extended period of time is not recommended. The information provided by the object recognition process provided by SURF supplied the required event detection needed to be used within security monitoring scenarios. These events can be clearly linked to operational controls used to secure objects within a secure environment to supply alerts to a user to enforce those controls. The work in this paper demonstrated that computer vision technologies, specifically SURF algorithms, can play a role in improving the enforcement of operational controls in secure computing environments. This step towards an automated security solution that can be used in diverse situations is clearly needed in an age where security of many assets, including human lives, is of utmost importance.

REFERENCES

[1] N. G. Carr, “IT Doesn’t Matter,” *Harvard Business Review*, vol. 81, no. 5, pp. 41–49, 2003.

[2] I. O. f. S. Commission and I. Electrotechnical, “ISO/IEC 27001:2005, information technology - security techniques - information security management systems-requirements,” 2005.

[3] ——, “ISO/IEC 13335:2004, Information technology - Security techniques - Management of information and communications technology security,” 2004.

[4] D. H. Harris, “How to Really Improve Airport Security,” *Ergonomics in Design: The Quarterly of Human Factors Applications*, vol. 10, no. 1, pp. 17–22, Jan. 2002.

[5] C. Colwill, “Human factors in information security: The insider threat Who can you trust these days?” *Information Security Technical Report*, vol. 14, no. 4, pp. 186–196, 2009.

[6] H. Bay, T. Tuytelaars, and L. Van Gool, “SURF: Speeded Up Robust Features,” *Computer VisionECCV 2006*, vol. 3951, no. 3, pp. 404–417, 2006.

[7] D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[8] J. Eichhorn and O. Chappelle, “Object categorization with SVM: kernels for local features,” *Biological Cybernetics*, vol. 190, no. 137, pp. 365–382, 2004.

[9] B. Bhanu and S. Lee, “Adaptive image segmentation using a genetic algorithm,” *Systems, Man and Cybernetics,*, 1995.

[10] K. Hammouche, M. Diaf, and P. Siarry, “A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation,” *Computer Vision and Image Understanding*, vol. 109, no. 2, pp. 163–175, 2008.

[11] D. Lowe, “Object recognition from local scale-invariant features,” *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pp. 1150–1157 vol.2, 1999.

[12] P. Viola and M. Jones, “Robust Real-time Object Detection,” *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2001.

[13] T. B. Terriberry, L. M. French, and J. Helmsen, “GPU Accelerating Speeded-Up Robust Features,” *Proc of 3DPVT*, no. 2, pp. 355–362, 2008.

[14] D. Nister and H. Stewenius, “Scalable Recognition with a Vocabulary Tree,” *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Volume 2 CVPR06*, vol. 2, no. c, pp. 2161–2168, 2006.

[15] R. Fergus, P. Perona, and A. Zisserman, “A Visual Category Filter for Google Images,” *Proc 8th European Conf on Computer Vision*, vol. 3021, pp. 1–14, 2004.