Interactive Computer Vision System: Recognizing Animals on the Savannah in Aalborg Zoo

Daniel Valsby-Koch

School of Information and Communication Technology
Aalborg University, Denmark
dvalsb09@student.aau.dk

Abstract. This paper examines the problem of creating an interactive computer vision system for recognition of animals on the Savannah in Aalborg Zoo. The current information signs at the zoo is insufficient and the information should be audible as well. With focus on simple pixel based features and a simple KNN classification, a complete system is proposed and tested for verification. The test is performed by test participants testing the system as a complete finished system working in Aalborg Zoo ...

1 Introduction

Ever since the Roman empire, or maybe even before that, the human people has strived towards being entertained. Today one is able to be entertained and informed in many ways including museums and zoos. When visiting these places, one is often greeted by stationary information signs, where the information is limited to a few languages and only contain "old news". Another limitation is the unavailability for people who cannot read e.g. children. In this paper I focus on developing and testing a complete vision-based system, that is able to recognise an animal in the zoo and thereby present some relevant visual and audible information to the user. The system strive to fulfil following demands: information available for children, regular update of information and news, educational entertainment.

Of course some or all of these demands are already fulfilled by other systems. An example is the usage of audio tour guides in The Metropolitan Museum of Art [1]. Here the visitor is able to rent an audio system, where they just need to play the correct information at the right spot. Another solution is the usage of smartphone apps. The London Zoo have an app where one is able to see when the next feeding is scheduled, where the animals are placed using the GPS in the smartphone and read some information about the animals in the zoo [2]. An example of an educational entertainment system is a project from Aalborg University in collaboration with Koldinghus castle [3]. Here the visitor is equipped with an iPad and is able to play around in the castle and hear

about the old time. This is done by recognising some placed game cards with the camera from the iPad.

The developed solution presented in this paper, is a stationary system with a camera, screen and computer being able to extract pixel based features and use these for recognising up to five different animals on the Savannah in Aalborg Zoo, Denmark. Thereby, the user is able to receive information both visually and audible about the interesting animal. The solution is a complete system meaning that it is wrapped by a user friendly interface, with no needed human pre instruction to work.

2 Previous Work

Recognising animals in the wild has been an area of great interest for several years. Biologists want to monitor the wildlife and thereby collect information about the population, movement and behaviour of specific animals [4]. A common situation includes a camera trap, which is either recording constantly or only when movement is observed [5]. Another common factor is the usage of video data. The camera trap is often stationary, which makes it possible to do background subtraction as a part of segmenting the animal[6]. Other approaches make the user mark the location of the animal by hand [7]. The segmentation part of the previous work is mostly not usable for this method,as the camera is not stationary. In addition, the environment is less controllable when being in the real life compared to a zoo. Therefore, some of the methods are more advanced than needed for this system.

3 Proposed System

The solution is a complete system meaning, that it contains both a method for recognising the animals and a user friendly interface for the user to interact with the system. Therefore, considerations for both the computer vision and the interactive system are needed. The solution consists of the following hardware parts:

- Digital Single-Lens Reflex (DSLR) camera with a 200 mm lens
- Prototype stand with a mounted screen and the camera, enabling the possibility to rotate the camera and screen 360° around itself. The stand is also able to tilt up and down, covering the whole Savannah
- Laptop running the software and connected to both the camera and screen
- Numpad to control the program

The software contains the following parts:

- Simple segmentation
- Pixel based feature extraction
- K-Nearest-Neighbour (KNN) classification
- Result evaluation
- Graphical User Interface (GUI)

3.1 Segmentation

The idea about the solution is to develop a possible simple system to meet the needs. There is overall two ways to do the segmentation. Either a complete segmentation of the animal, which makes it possible to use the shape and size as features. Or a partly segmentation, where only some of the animal is segmented, and a pixel based feature approach is preferred. Keeping the simple approach in mind a user dependent partly segmentation method is implemented. The user is told to ensure that a specific amount of the middle is containing an animal and no background. To improve the recognition rate, not only one square is cropped from the image. A field of 3 by 3 squares (15 by 15 pixels) is cropped from the image, and each square is used for individual feature extraction and classification. By using this approach, the system is able to recognise the animal based on only some of the squares. Figure 1 visualises how the squares are cropped from an image.

Fig. 1. Left image is correctly captured with all squares being inside the animal. The right image is wrong captured, where only some of the squares are inside the animal.

3.2 Feature Extraction

Only a partly segmentation is performed, which means the features used should be pixel based, as there is no information about the entire animal. When looking at the five animals, two categories of features seems to come in handy. The five animals are separable by the human mind when comparing their color and texture. Therefore, seven features are used to recognise the animals namely: hue, saturation, horizontal Sobel, vertical Sobel, mean, variance and Canny Edge Detection.

The hue and saturation is easily extracted by converting the RGB color to HSV. The value is not used because of the probability of variation in light

intensity. The conversion is performed by the following equations [8]:

$$H = \begin{cases} \frac{G-B}{V-min\{R,G,B\}} \cdot 30°, & \text{if V=R and G} \geq \text{B;} \\ (\frac{B-R}{V-min\{R,G,B\}} + 2) \cdot 30°, & \text{if G=V;} \\ (\frac{R-G}{V-min\{R,G,B\}} + 4) \cdot 30°, & \text{if B=V;} \\ (\frac{R-B}{V-min\{R,G,B\}} + 5) \cdot 30°, & \text{if V=R and G} < \text{B;} \end{cases} \qquad H \in [0°, 180°[\quad (1)$$

$$S = V - min\{R, G, B\} \qquad S \in [0, 255] \quad (2)$$

$$V = max\{R, G, B\} \qquad V \in [0, 255] \quad (3)$$

The horizontal and vertical Sobel is computed by applying the Sobel kernels. The sobel kernels are seen in Figure 2. This feature describes the vertical and horizontal edges in the image.

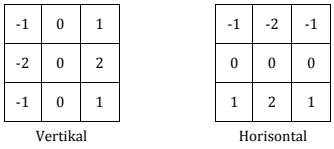

Fig. 2. Left is the vertical kernel and right is the horizontal kernel.

The mean and variance of the image are also used as features. The mean is actually the average gray scale value of the image. The variance describes whether there is a big difference between the lowest pixel value and the highest. Both values are calculated by the following equations:

$$\mu_x = \frac{1}{N} \sum_{n=0}^{N} x_n \quad (4)$$

$$\sigma^2 = \sum_{n=0}^{N} \frac{(x_n - \mu_x)^2}{N} \quad (5)$$

Finally, the Canny Edge Detection is used to find all edges in the image and give a representation of the animals texture. Briefly, the Canny Edge Detection is functioning by using the Sobel Kernels to find the maximum gradients. After that a double threshold is performed followed some post processing.

A plot of the feature vectors is seen in Figure 3 and 4.

3.3 Classification

One of the most simple classification methods is the KNN, which is why this is chosen classification method. A KNN algorithm tries to classify a feature vector

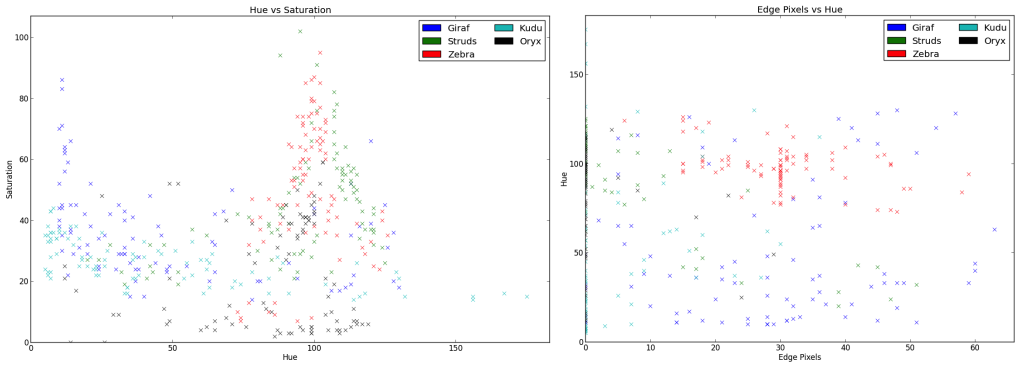

Fig. 3. Left is a plot of the hue vs the saturation. The right plot is Canny Edges vs hue.

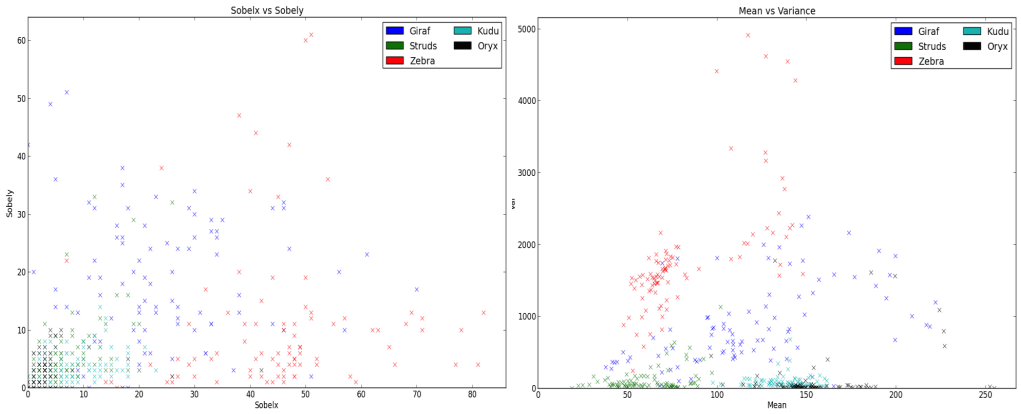

Fig. 4. Left is the horizontal Sobel (sobelx) vs the vertical Sobel (sobely). Right image shows the plot of the mean and variance.

to a given class, by calculating the distance from the vector to the k nearest training vectors. Mathematically, the KNN rule tries to estimate the a posteriori probability $P(\omega_i|\mathbf{x})$, which is seen in 6.

$$P_n(\omega_i|\mathbf{x}) = \frac{p_n(\mathbf{x},\omega_i)}{\sum_{j=1}^{c} p_n(\mathbf{x},\omega_j)} = \frac{k_i}{k} \tag{6}$$

where: $p_n(\mathbf{x},\omega_i) = \frac{k_i/n}{V}$ is the estimate for the joint probability $p(\mathbf{x},\omega_i)$, where V is a cell of volume around $\mathbf{x}$, including k_n out of n samples [9]. A $k = 3$ is chosen based on a test on the training data.

3.4 User Interface

When developing an interactive system, it is very important to focus on the user interface and the usability. The GUI includes six screens being: welcome screen, live view screen, waiting screen, animal information screen, playing sound screen and an animal not recognised screen. The combination of these screens is the final user interface. An example of the GUI is seen in Figure 5.

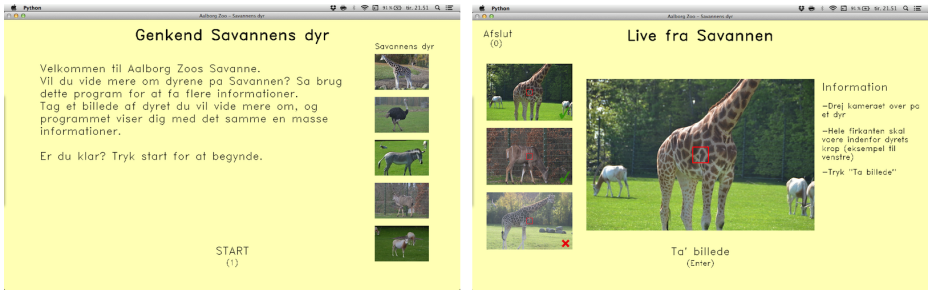

Fig. 5. Left image is the welcome screen with information about the system. Right image is the live view screen shown to the user. In addition to the video feed from the camera in the middle, this screen also contains a written and visual exemplification of how to use the system. Note the red square, which is the minimum needed to be filled with an animal body.

A final system should contain a touch screen for interaction with the program, which in this case is handled by a numpad.

4 System Test

Several tests are conducted to verify the recognition rate of the system. Both theoretical test by 2-fold cross-validation and real life prototype testing with test participants is included in the system test. Another aim for the prototype test is to see if the participants are able to use the system correct without any human instructions. Firstly, the training data is briefly explained.

4.1 Data

As the classification method is a supervised classifier, some labelled training data is needed beforehand. The dataset consist of 310 images distributed the following way: giraffe 78, ostrich 44, zebra 58, kudu 46 and oryx 84. This dataset is used for training the classifier before recognising animals at the prototype test. For the 2-fold cross-validation, a segregation of the dataset into two equally large datasets is needed. The two dataset is denoted d_1 and d_2.

4.2 2-Fold Cross-Validation

The 2-fold cross-validation is made in two test. The first test uses d_1 as training data and d_2 as validation. The second test switched the two dataset, using d_2 as training and d_1 for validation. The result from this is seen in Table 1.

	Correct recognised	Wrong recognised	Not recognised
Test 1	83,23%	13,55%	3,23%
Test 2	90,32%	5,16%	4,52%
Average	86,78%	9,36%	3,88%

Table 1. Results from the 2-fold cross-validation. Correct recognised means the animal is recognised as the animal it is. Wrong means an animal is recognised as a wrong animal. Not recognised means an animal is classified as being non-animal.

4.3 Prototype Test

The prototype is tested by 16 participants spanning from the age of 8 to 71. The participants are told to use the system as described on-screen and try it as long as they want. Afterwards they are asked to fill a questionnaire concerning their usage of the prototype. The test data from this ends up with three different results, depending on the circumstances:

1. All images captured by the participants are used for the results
2. Only correct captured images (red squared filled with animal) are used for the results
3. Images captured by the author are used for the results

The results from these three tests are shown in Table 2.

	Pics	Corr. recognized	Wrong recognised	Corr. rejected	Wrong rejected
Test 1	106	34,91%	29,25%	6,69%	29,25%
Test 2	60	58,33%	15,00%	0,00%	26,67%
Test 3	21	66,67%	14,29%	0,00%	19,05%
Average	62,33	53,30%	19,51%	2,23%	24,99%

Table 2. The results from the prototype test. The data is divided into three different dataset, as explained earlier.

Questionnaire: The following is to be highlighted from the questionnaire:

- Most of them thought the prototype was easy or very easy to use
- Most of them think the program recognised the animals fairly or better
- Many thought a zoom function would be great
- All text should be audible
- Entertaining to interact with the animals

5 Discussion

The 2-fold cross-validation showed some good results with the simple feature extraction and classification implemented. An average recognition rate of 86,78% is satisfying, and seems like a very good rate when focusing on the simplicity of the system. A reason behind the high recognition rate might be the training data. Some of the data is quite equal, because the data acquisition were only divided into three days. The good theoretical recognition rate is not retained at the prototype test, sadly. The prototype test shows significant lower recognition rate being 34,91% at the worse. It improves quite well (58,33%) when only using correct captured images, and increases even more when the author uses the system (66,67%). The first thoughts about this is the usability of the system. Looking at the lowest recognition rate, it seems that the user either don't know how to use the system or can't use it correct. It is actually only 56,60% of the captured image that were correctly captured. By monitoring the test participants some problems are revealed:

- Some of them, mostly the children, had problem with the control of the prototype. The prototype is too unhandy for some
- A lot of them didn't understand, forgot or ignored the fact that the animal should be bigger than the red square. This is the main reason for the wrong captured images
- Sometimes the animal were too close, sometimes it were too far away
- Sometimes the animal moved quickly while the image was captured, resulting in some very blurred images
- The sun created shadows at some of the animals, which resulted in edges on animals normally not containing edges
- Sometimes an image of the head was captured. The training data contains mostly images from the body of the animal

Looking at these problems both an improvement to the GUI, usability and recognition module is preferable. The GUI needs to be updated such that the user remembers to capture a correct image. Some kind of a feedback would be great, telling the user whether the animal is the correct size for recognising it. The system is only a prototype, and it is expected to be a little unhandy. An important consideration for a potential system is to make it as easy to navigate as possible. Regarding the recognition module, a lot of improvements are possible. A more thorough evaluation of the features might reveal some problems with the chosen

features and other features might be better. A big problem is the scaling of the animal. It is very dependent of the distance between the animal and the camera, and a scale independent solution like SURF features might reveal some good results. The classification is also possible to optimise. Right now only a KNN is used for the classification. Depending on the distribution of the feature vector, a dimensionality reduction of the feature space could come in handy. Some more advanced classification methods like Support-Vector-Machine might increase the recognition rate.

Another important improvement is the addition of several languages and reading of all the written text. This is only a small programming work, but will extend the usage of the system a lot. Also the missing implementation of autofocus on the camera is to be corrected.

6 Conclusion

The conducted research proved, that the proposed system is partly usable. The upper recognition rate of 86,87% indicates that it is possible to make a animal recognition system even with a simple implementation. Several possible optimisations are discussed and indicates where the optimisation should be made. The results from the prototype test in Aalborg Zoo revealed, that a system like this is desired by people spanning from the age of 8 to 71 years. This means that it should be interesting for entertainment parks like zoos to investigate this field.

7 Acknowledgements

The author would like to thank Aalborg Zoo for participating in the research and make their zoo and animals available. Thanks also to Thomas B. Moeslund for supervising the project and to Claus B. Madsen for help with the prototype stand.

References

1. The Metropolitan Museum of Art: Audio Guide, http://www.metmuseum.org/visit/plan-your-visit/audio-guide. (2014) Downloaded: 29-05-2014.
2. ZSL London Zoo: London Zoo App, https://itunes.apple.com/gb/app/london-zoo/id394794108?mt=8. (2014) Downloaded: 29-05-2014.
3. Claus B. Madsen: Koldinghus Augmented: The Walls Remember, http://vbn.aau.dk/files/58116502/ProjektBeskrivelseKoldinghusAugmentedv20.pdf. (2011) Downloaded: 29-05-2014.
4. Rowcliffe, J.M., Carbone, C.: Surveys using camera traps: are we looking to a brighter future? Animal Conservation **11** (2008) 185–186
5. Jens Wawerla, Shelley Marshall, G.M.K.R., Sabzmeydani, P.: Bearcam: automated wildlife monitoring at the arctic circle. Machine Vision and Applications **20** (2009) 303–317

6. Pooya Khorrami, J.W., Huang, T.: Multiple animal species detection using robust principal component analysis and large displacement optical flow. Beckman Institute, University of Illinois - Urbana Champaign (2012)
7. Mayank Lahiri, Chayant Tantipathananandh, R.W.D.I.R., Berger-Wolf, T.Y.: Biometric animal databases from field photographs: identification of individual zebra in the wild. ICMR '11 **6** (2011)
8. Moeslund, T.B.: Image and Video Processing, 2nd Edition. Number ISBN: 978-87-992732-1-8. Aalborg University (2009)
9. Daniel Valsby-Koch, Flavien Cortella, J.V.D., Nielsen, L.N.: Speed sign recognition. Aalborg University (2012)
10.