Identifying Animal Herds in Aerial Images Captured by a Moving Vehicle

Andreia Teixeira and David Pereira
FCUP – University of Porto
Portugal
EMail : andreiasofiat@hotmail.com, dpereira@liacc.up.pt, mcoimbra@dcc.fc.up.pt

***Abstract*—The contents of this paper relate to a first approach towards the identification of animal herds in images captured by an aerial moving vehicle. In particular, we will present some experimental results about the usefulness of basic color properties in detecting and counting individuals in herds of Gnus. We discuss out approach and the results obtained. We also point future work directions in order to obtain more robust results.**

I. Introduction

Nowadays, one of the interesting subjects that Ecology-related research groups address is the understanding of the group behavior of animal herds. An optimal solution to get this understanding would be to attach to each animal of the herd some kind of wireless sensor which would monitorize the overall (interesting) activity of each of these animals. After collecting this information, one could understand how the individuals interact with each other and how that influences the overall behavior of the herd. However, this seem quite difficult (if not impossible) task to do, since it would imply hardware and logistic costs of great proportions (veichle renting, capturing each element of the heard, recollect the sensors once the task was over, etc.).

What seems to be the best alternative to understand the group behavior is to apply a Computer Vision approach to the problem. Using visual information (pictures and videos), one could obtain information in an automatic way. The first step towards this approach is data collection and processing.

Data collection and processing is a hard job since currently thousands of images need to be captured and individual animals identifies by hand, as a pre-processing stage for further studies. This is a huge workload for human researchers and an important bottleneck of all research. Automation using image processing and computer vision methodologies can be a vital factor in the success of projects related to the understanding of group behavior of animal herds.

In this work we are interested in giving the first steps towards finding gnus in images taken from an aerial moving vehicle and also to count them (estimate the overall number of elements in the herds). No video was recorded in this phase of the research. Having these goals in mind we present in this paper some experimental results of simple color analysis on the set of available pictures. As we will see further ahead in this paper, we got quite different results depending on the color properties and classification tasks we experimented.

It is worth noticing that this work focus on classification of samples extracted from the pictures available. We are interested in classifying these samples as being part of regions in an image (water, sky, land, gnus) and also to classify samples just as being gnus or not, independently of what kind of region they belong. In our understanding of the problem, if the simple color properties we have studied behave relatively well under a *machine learning* (classification) perspective [1], these properties should be further explored and combined with other Computer Vision methods to get accurate classifications and go to a next stage which would be developing a software system capable of identifying and give estimates of the number of elements in heard without much intervention from ecology specialists.

This paper is organized as follows: in Section 2 we give an overview about the characteristics of the images we have available, in Section 3 we detail the research methodology used in this work, in Section 4 we present the results obtained and in Section 5 we make some conclusions about the state of the work and point future work to address this problem.

II. Pictures available

We will now give a simple overview of the type of images one had access to perform our experiments. We have available near to 100 aerial images, captured by a moving vehicle in the African Savannah. These images have no sequence and form a quite heterogeneous set. For presentation proposes, we devided the overall set of images in three classes: *near*, *far* and *very far*. We present an example of each kind of picture in Figure 1, Figure 2 and Figure 3, respectively. It is easy to notice that what one can visually classify as a Gnu changes a lot from image to image.

III. Research methodology

As we have already stated, we consider only simple color properties to try to identify and count gnus on images similar to the ones presented in the previous section. We will now describe the methodology we have choosen to measure the usefulness of these properties.

Our methodology considers two kinds of data:

1) average and standard deviation of Red, Green and Blue components.
2) the concatenation of the Red, Green and Blue histograms.

Fig. 1. Close range photos.

Fig. 2. Far range photos.

The first kind of data was used to classify parts of an image as belonging to one of the following classes:

- *Land :* savannah ground;
- *Gnus :* gnu body;
- *Water :* water from lakes and small rivers;
- *Sky :* sky.

In terms of goals, this data was used to distinguish between parts of an image which contain gnus and parts of the image which do not contain them. There is no intention of individual Gnu identification and counting in this sub-task. Figure 5

Fig. 3. Very far range photos.

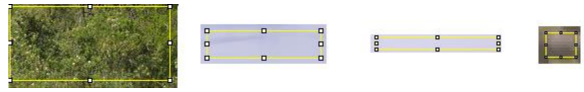

Fig. 4. Samples of ground, ground with gnus, water and sky

R	SDR	G	SDG	B	SDB	Class
95.701	30.569	101.766	30.339	52.007	25.179	Land
128.775	25.026	116.462	21.617	106.788	20.038	Gnus
195.921	4.632	197.762	4.127	222.787	2.937	Water
215.208	1.83	212.555	1.952	234.53	1.727	Sky

Fig. 5. Table with the numeric data representing the color properties corresponding to Figure 4 samples.

represents the kind of data extracted from the samples of Figure 4. We have extracted data from near to 150 picture samples. The second part of our methodology considered the same properties of images, plus the size of the sample extracted from the original picture. In this part we considered the following classes, which characterize a sample by the number of gnus in it:

- *GnuN_0 :* sample without gnus;
- *GnuN_1_5 :* sample with 1 to 5 gnus;
- *GnuN_5_10 :* sample with 6 to 10 gnus;
- *GnuN_10_15 :* sample with 11 to 15 gnus;
- *GnuN_15 :* sample with more that 15 gnus.

In the third and last sub-task of our experiments, we considered the RGB histograms of samples, and made a simple division in two classes, *Gnu* and *NotGnu*, whose meaning is evident. However, for extracting the concatenation of the RGB histograms, we developed a simple script, written in Python which used the PIL module and interacts with Weka command line facility [].

In terms of classification we took a rather naive approach. We just tried Weka's classifiers and made a rank of the one which best fit our classification needs. As best classifier we got J48 and that was the one we used in this work, altough experiments with other classifiers such as Naive Bayes and Linear Discriminat were tested.

IV. Tools

We now briefly talk about the tools we have used in our experiments. We used the following:

- ImageJ : [4] it is a public domain Java image processing program. The main advantage of this tool is that it has an automatic way to produce numeric values that interest us. Figure IV shows an example of this automatic extraction of measures.
- Weka : [6] popular suite of machine learning software written in Java, developed at the University of Waikato. We use it here to An alternative would be to use the Orange Framework, an Weka-like framework fully implemented in Python.
- Python : [5] is a simple and powerfull script programming language. We used it to write scripts which serve as glue code to connect the previous tools. We selected Python maily because it is flexibile and easy for program

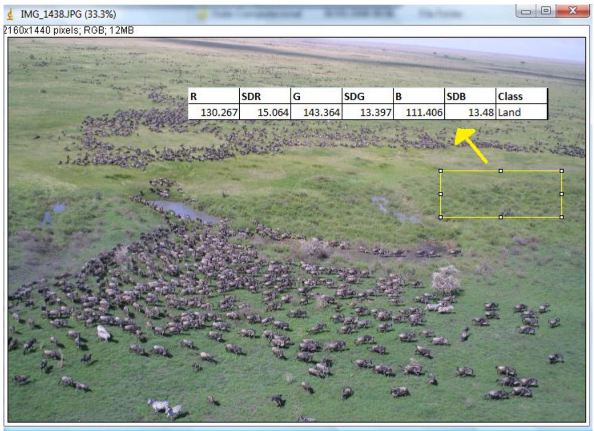

Fig. 6. Automatic extraction of measured data from an image, using ImageJ.

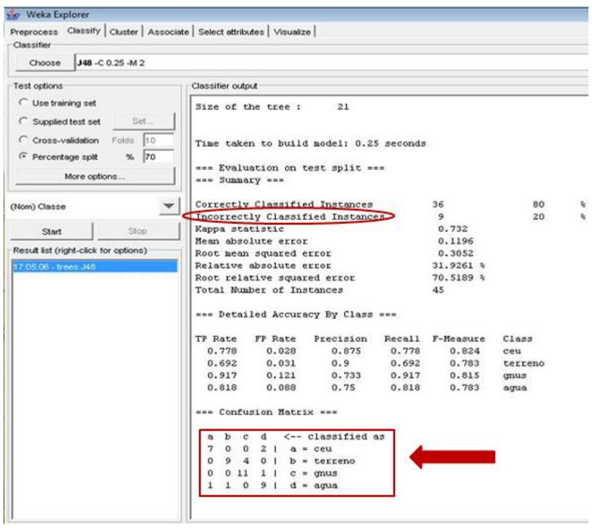

Fig. 7. Classification using Weka.

development, plus the availabilty of the image processing library PIL [3].

V. Experimental Results

We have obtained different result accuracy in the three subtasks describe in Section 2. The first one (classifying parts of an image as being either land, land with gnus, water and sky) provided us the best results, followed by the classification of individual gnus by histogram, and in last place came the gnus' counting by selecting samples with different number of gnus.

The classification of samples as being either land, gnus, water and sky produced classification results of 80% of correctly classified samples, using the J48 and the Linear Discriminant. In particular for a small test dataset the classification resulted in the algorithms correctly classifying all the samples of gnus and land. However it classified in a wrong way two instances of sky and water. Figure **??** shows this *"confusion"*. We believe that solving this problem is directly connected to identifying the line of the horizon in the images we process. That way we know what is water or sky based on its position (above or below the horizon line) in the given image.

R	DPR	G	DPG	B	DPB	Classe	With R	Reality
199.117	4.482	206.538	3.851	250.65	2.869	?	Sky	Water
177.062	1.292	168.4	1.037	181.097	1.186	?	Water	Water
150.643	1.386	143.461	1.626	158.72	1.528	?	Water	Water
156.743	3.733	155.983	3.519	169.529	3.986	?	Water	Water
85.131	23.746	83.357	20.742	94.06	20.346	?	Gnus	Gnus
207.345	3.354	211.484	1.982	242.344	2.936	?	Sky	Sky
148.859	2.046	154.675	2.006	176.562	2	?	Water	Water
125.258	5.223	128.959	4.661	78.141	4.953	?	Land	Land
220.863	4.211	225.899	2.516	255	0	?	Sky	Sky
83.818	11.387	83.46	10.14	24.426	6.238	?	Land	Land
122.937	26.067	112	22.743	129.032	22.51	?	Water	Water
94.49	16.428	90.8	14.081	104.55	14.115	?	Water	Water
100.621	17.631	97.718	14.969	110.596	16.238	?	Water	Water
121.542	24.54	121.27	23.356	60.127	20.319	?	Land	Land
197.13	2.048	198.671	1.692	222.785	1.68	?	Sky	Sky
103.795	20.084	94.63	16.673	98.106	16.392	?	Gnus	Gnus
121.481	7.387	124.843	7.256	68.388	7.546	?	Land	Land
149.117	5.306	150.37	5.584	185.359	7.035	?	Water	Sky
209.567	12.479	208.344	7.133	222.004	6.896	?	Sky	Sky
102.929	25.907	82.762	21.371	82.571	23.355	?	Gnus	Gnus

Fig. 8. Classification of new test instances.

```
Time taken to build model: 0.54 seconds
Time taken to test model on training data: 0.11 seconds

=== Error on training data ===

Correctly Classified Instances         240               99.1736 %
Incorrectly Classified Instances         2                0.8264 %
Kappa statistic                          0.9835
Mean absolute error                      0.0161
Root mean squared error                  0.0897
Relative absolute error                  3.2199 %
Root relative squared error             17.944  %
Total Number of Instances              242


=== Confusion Matrix ===

   a   b   <-- classified as
 120   1 |   a = GNU
   1 120 |   b = NOTGNU


=== Error on test data ===

Correctly Classified Instances          36               69.2308 %
Incorrectly Classified Instances        16               30.7692 %
Kappa statistic                          0.3846
Mean absolute error                      0.3107
Root mean squared error                  0.5502
Relative absolute error                 62.1349 %
Root relative squared error            110.039  %
Total Number of Instances               52
```

Fig. 9. Classification of new test instances.

The results obtained using histogram concatenation were not so good as the previous ones. We just got near 69% of correct classifications. The best classifier was again J48. The result of automatically extracting histograms and classifying them is presented in Figure V.

In the case of the last experiment, which was an attempt to get an estimate on the number of gnus in a picture using only the average and standard deviation of the RGB componets, as in the first experiment. In this case we got a result of 62% correctly classified instances. However, the test instances contained as feature their area (in pixels) in the original image. Removing this feature, we get a disappointing 33% of right classifications. The analysis of this difference of result will be discussed in the next section.

At the current stage of this research no more experiments were conducted. In the next section we will present our interepretation of this results and point the next steps towards the goal of obtaining more robust classification results.

VI. Conclusions and Future Work

We will now draw our main conclusions about the work we have presented. This work uses a rather simple approach based on averages, standard deviations and concatenation of the histograms of the RGB components. We have used these properties as the features of a feature vector and looked at the problem from a pattern recognition point of view. We have obtained results which we consider as not being bad and which point that a great effort in exploring color properties of herds of gnus in the savannah will end up bringing the foundations for building automatic software systems to help ecology related groups have a better understanding og groupal behavior of herds.

We had pretty good results in identifying parts of images as belonging to one of the following classes: water, sky, gnu and land. The chosen classifier, the decision tree's J48, provided output which mainly failed when classifying water and sky. However, in principle this can be avoided in the future by augmenting the training dataset and pre-processin images by finding the horizon on images. In the later case, every part of the image which may look like water but that is above the horizon would be correctly classified as sky. The same would work for water if the test image is below the horizon.

The two other approaches were identifying samples of images as being or not individual gnus (using histograms) and estimate the number of gnus in a herd. Identifying gnus using histograms had a 69% of correct classifications, which is lower than 80% of classifying the parts of images as described in the previous paragraph. We conclude that using histograms did not brought anything new to the solution of the problem. Counting gnus by considering parts of the image with different quantities of gnus also provided 62% of correct classifications. However this rate was mainly granted by the feature representing the area of the gnu group. Therefore, at least for identifying gnus and counting them, the average and standard deviation seems the best descriptors for identifying gnus in the images we used in our research.

Identification of gnus as an automatic task requires more than the 80% we have obtained. More learning algorithms [1] should be tested and the better suited must me studied in order to try to understand the mathematical reason of their success and, after that, figure out how to improve them for our purposes. Also more advanced techniques for gnu identification should be explored such as hedge detections, template matching based on simple mathematical geometrical figures like elipses, and others [2]. We believe we took the first step towards a very interesting and promising project whose goal may help our understanding of the behavior of important species which share organized social group activity.

References

[1] Ian H. Witten and Eibe Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2002.

[2] David A Forsyth, Jean Ponce, *Computer Vision: A Modern Approach*, 2002.

[3] *Python Imaging Library (PIL)*, http://www.pythonware.com/products/pil/ .

[4] *ImageJ – Image Processing and Analysis in Java*, p://rsb.info.nih.gov/ij/.

[5] *Python Programming Language*, http://www.python.org.

[6] *WEKA - The University of Waikiato*, http://www.cs.waikato.ac.nz/ml/weka/.