Designing a Framework for Animal Identification

Hans Krijger
hglk@rucus.ru.ac.za

Greg Foster
g.foster@ru.ac.za

Shaun Bangay
s.bangay@ru.ac.za

Department of Computer Science
Rhodes University
Grahamstown, South Africa

Abstract

The conventional methods of animal identification can be replaced with a semi-automatic image analysis tool, which distinguishes individuals based on their unique markings. A flexible framework for the analysis must encompass a combination of relevant features with interchangeable animal-specific modules. Developing a Java-ImageJ plug-in alleviates routine functionality, but enforces some degree of conformity. Zebra photographs are used as the initial data under consideration. De-interlacing, adaptive thresholding, smoothing and sharpening are identified as beneficial pre-processing steps. Binarisation and sequential thinning are discussed as essential processing stages. Pattern extraction and matching is based on vectors relative to a manually defined region of interest. Provision for enhancing the system to allow fully automatic processing must be made.

Keywords: identification, matching, image processing, biometrics, region of interest, ImageJ plugin, zebra

1 Introduction

Recording animal movement is an important process [1] that provides wildlife management with valuable insight on the feeding habits, social interaction, and migration trends of a species. Various methods exist for tracking individual animals both in the wild and around game reserves. The animal can be tranquillised and fitted with a radio-transmitter, which gives constant accurate information on its whereabouts, but this is both expensive and intrusive. A simple identification tag may be attached instead, but this may require the animal to be re-captured every time information is gathered.

A subset of wild animals, including the cheetah [1], zebra [2], giraffe [10], elephant [11] and coelacanth [12], possess different forms of unique individual markings, similar to human fingerprints. These can be used to successfully identify individuals from photographs, which are obtained without direct contact, and from a non-intrusive distance. However some previous experience with the animals is necessary to distinguish them. The work is both tedious and time-consuming, especially as the population size increases. All of these techniques require trained professionals.

By automating the examination of photographic data, an animal specific expert is no longer essential to the identification process. Additional benefits include lower costs, faster matching, scalability, and system portability. An automated system must be simple to use and generalised so that any training required is applicable to identifying highly varied species.

2 Overview

This paper provides a description of the framework developed for semi-automatic identification through the analysis of digital photographs. The animal under consideration is the zebra *(equus burchelli),* although the system may accommodate other species. Various well defined image pre-processing techniques are discussed in reference to this system, including de-interlacing and adaptive thresholding, as well as issues faced in developing the tool as a plugin for ImageJ - an image processing package written in Java [7]. Two image processing techniques, binarisation and thinning, are evaluated in the context of identification. This paper also describes the design for a generic pattern extraction mechanism, including preparatory steps for matching. Finally outstanding work is listed.

3 Related Work

The benefits of computer aided identification are documented by Kelly [1] with specific reference to a cheetah matching system. Other unique animal patterns such as the giraffe are also mentioned.

Peterson [2] discusses zebra stripes in detail, using the lateral, cervical, leg and scapular stripes for identification.

Biometrics and issues surrounding fingerprint recognition are well documented by Roddy and Stosz [3], Maio and Maltoni [4], and Castleman [5]. This work has a direct relationship with the pattern extraction and matching performed in this study.

The complete ImageJ package, with source listing and documentation is available free online [7].

4 ImageJ and Java

The framework described here has been named PEMI (Pattern Extraction and Matching Interface), and was developed as a plugin for ImageJ. This is an open source image processing package written in Java. ImageJ enables rapid development of image analysis tools by providing routine file handling and image processing functionality, as well as many common image filters, transformation operations, and various analysis plugins. ImageJ is widely used and supported, most notably in the medical imaging field. Through open source development enhancements and fixes are made available immediately, and owing to the portability inherent in Java, it may be run on a variety of platforms as a standalone application or as a browser plugin.

While the development process is both efficient and portable, there are some drawbacks. Firstly, the package is written for the Advanced Windowing Toolkit (AWT), thus any new graphical features provided by Swing are obsolete. This is also true of the Java Advanced Imaging (JAI) and Java 3D toolkits. Secondly, since private data members in the ImageJ base classes are inaccessible, some restriction is placed on coding practice. Personal experience has shown that the author of the software can rectify this problem in future releases of the software by designating the required variables protected or public, as is appropriate. Thirdly, it may be necessary to duplicate much existing code in order to modify an algorithm slightly. This must be weighed up against writing a standalone analysis tool with the associated housekeeping code. While some duplication was necessary, PEMI does not warrant an independent system.

5 Image Pre-processing

The first stage of the recognition process deals with preparation of the image itself, rather than the information contained within the image.

5.1 Quality and clarity issues

An identification system is likely to deal with photographic images captured from a video stream or digital camera. If the video camera movement is too fast relative to the shutter speed interlaced frames may appear blurry (Figure 1a). This is due to misalignment of the scan lines within the image, and typically occurs when a high zoom is employed. In this situation even the effect of a slight breeze becomes amplified, so when a frame is captured one set of scan lines does not mesh with the other because the camera has moved too quickly. De-interlacing corrects this problem, producing a clear image (Figure 1b). It is often an important pre-processing step.

There are two de-interlacing techniques in PEMI. The first discards one set of scan lines and duplicates the other set. The second (Figure 1b), creates an average from each pair of scan lines. These methods produce visually identical results, although one may prove more suitable during the matching stage.

(a)

(b)

Figure 1: The effect of de-interlacing. (a) Original (b) De-interlaced using an average

Loss of image clarity may also occur due to poor lighting conditions (section 5.3), low camera or film quality, or quality loss during medium transfer. Correcting these problems involves two key steps: removing noise and enhancing edges. The smooth and sharpen filters within ImageJ are used to enhance the image sufficiently. These are 3×3 convolution filters, which significantly improve the results of binarisation (section 5.3) and thinning (section 5.4) when applied correctly. These two steps are the cornerstone of the matching process.

5.2 Region of interest (ROI)

The ROI defines an area of the image which contains the information to be extracted (Figure 2). In this instance we examine the lateral stripes of the zebra as these show the most variation between individuals (pers. ob.). Having a manually defined ROI with known points allows for flexibility, most notably with reference to alignment. Differences in alignment of both the animal and camera can be corrected through knowledge of the relationship between specific points, such as the points defining the back and front legs, which should be aligned on the *x*-axis. This alignment falls under the outstanding work, however, and shall not be discussed further.

PEMI requires a ROI to be defined before performing thinning (section 5.4) as this is a processor intensive operation, typically requiring 16 iterations through each pixel under consideration. It is important that the ROI does not vary greatly between images, in order to standardise thinning and later extraction as much as possible. For this reason every image-specific ROI is directly associated with the image file. It is also persistent, so that the ROI need only be defined once.

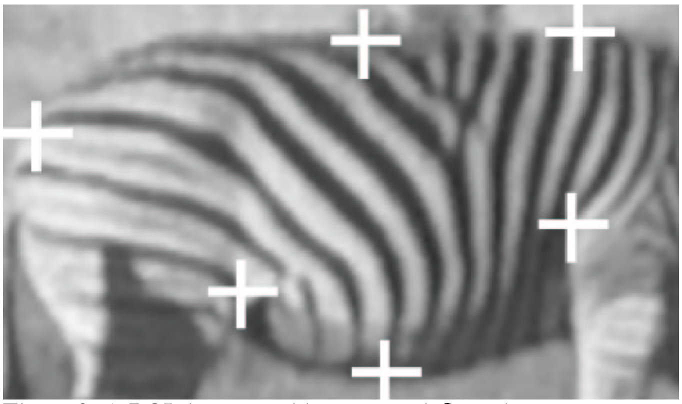

Figure 2: A ROI demarcated by crosses defines the area containing lateral stripes.

To provide persistence, the ROI (in the form of absolute integer co-ordinates) is stored within the image file itself. ImageJ supports tiff images and stores pixel data after byte offset 768. This typically provides 615 bytes of free space. The 4 byte integer co-ordinate pairs are stored after the file header information. The type of ROI under consideration is also recorded using a 7 digit identification number. This identification number is specific to the animal species as well as the area under consideration.

Defining the ROI is the only significant manual step in the identification process, and is unique to each image. Thus, having a persistent ROI provides for the possibility of fully automatic shell processing.

5.3 Image binarisation

PEMI is a generic framework for matching patterns. All of these patterns, while varying in size, shape and colour, consist of lines, edges and points. A useful way of enhancing these features in images captured under different circumstances is binarisation [8]. If an image is converted correctly to black and white such that the relevant edges or points can be distinguished from the surrounding data, the artefacts caused by poor lighting or shadows can be removed effectively. Such a method has advantages. Images captured from video, still photography, or scanning, with different lighting conditions, resolutions, colour balances, noise and alignments have at this point been standardised. Using de-interlacing we can remove the effects of video inter-frame movement, and using sharpening and smoothing filters we can effectively negate noise or resolution problems. With the correct binarisation technique we may be able to eliminate the effect of bad lighting and shadow, as well as removing colour imbalances and any brightness/contrast problems.

Adaptive thresholding [3, 4, 5] is the method used in PEMI. Firstly, the image is tiled into square blocks of size M^2 pixels. Next, every block is examined twice: once to determine the average intensity of pixels within that block, and a second time to compare each pixel intensity against the average. If the intensity is higher than the average that pixel is painted white, otherwise it is painted black. Because each pixel is compared only to its nearest neighbours, local effects such as light and shadow variation are removed. The tile size directly impacts a correct translation to binary (Figure 4) and in turn is related to the pixel size of the features under consideration. Currently the choice is manual, although it may be possible to determine an exact numerical relationship between the two.

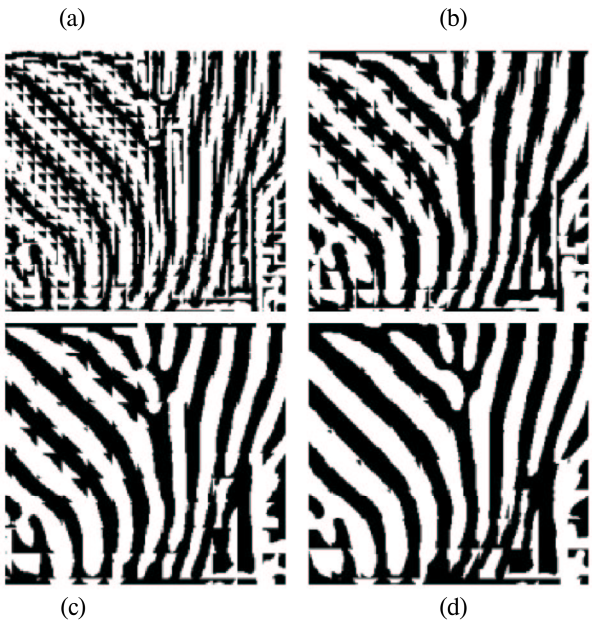

Figure 4: The effect of adaptive thresholding with different tile sizes. (a) M=10 (b) M=16 (c) M=22 (d) M=26

5.4 Skeletonisation by thinning

Thinning is well documented [3,5] and consists of iteratively removing pixels from binary shape boundaries. Both directional sequential and data parallel methods exist [6].

We have used the method outlined by Sonka et al. [8] which tests 8 boolean conditions on the 8 nearest neighbours of every pixel. If any of these conditions hold the pixel is removed. This is a form of erosion with structuring element $L_{(i)}$, and typically requires between 12 and 20 iterations before idempotency (or stability) is reached.

L_1 and L_2 are given below, the remaining 6 matrices can be found by successive rotations through 90°. A value of 1 refers to a black pixel, 0 to a white pixel, and * refers to either.

$$L1 = \begin{bmatrix} 0 & 0 & 0 \\ * & 1 & * \\ 1 & 1 & 1 \end{bmatrix} \quad L2 = \begin{bmatrix} * & 0 & 0 \\ 1 & 1 & 0 \\ * & 1 & * \end{bmatrix} \ldots L8$$

After thinning all structures have been reduced to one pixel wide. Correct skeletonisation must conform to the principle of homotopy so as to preserve the connectedness of each pixel, and in turn of each structure.

The benefit of reducing a pattern to one pixel wide is that it becomes simple to trace the resulting skeleton and construct a map of vectors with which to represent and thus match the pattern. The algorithm is processor intensive, so thinning is only performed over the region of interest already defined within the image. Generally this is a much smaller area and so can be processed much faster. When dealing with a polygonal ROI we simply form a bounding rectangle over which the sequential thinning takes place (Figure 5), but the ROI shape itself is not changed.

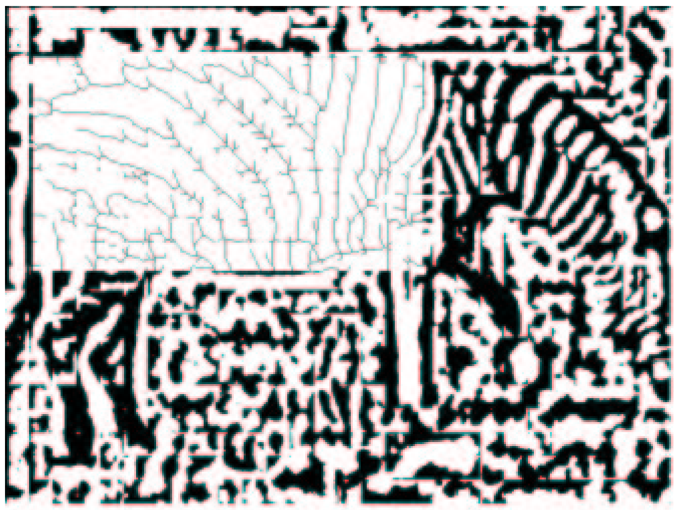

Figure 5: The lateral stripes within the ROI have been reduced to a homotopic skeleton.

The rest of the image is discarded (Figure 6). The ROI co-ordinates are changed to take this into account but they still designate the same area.

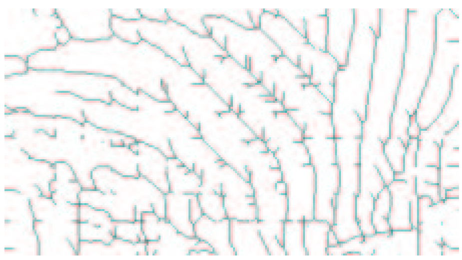

Figure 6: This skeleton represents the pattern to be extracted and matched in the next phase of analysis.

6 Pattern Extraction Foundations

As mentioned previously, PEMI enhances points, lines and edges in the initial stage because these are sufficient for representing and matching varying animal patterns. Zebra stripes will be represented directly by a series of vectors relating to the one-pixel wide contour maps produced by homotopic skeletonisation (Figure 6). This is generally true for lined patterns, such as elephant skin or other similar surfaces that consist of a number of ridges and valleys, similar to human fingerprints.

6.1 Quantitative vs. qualitative techniques

There are different possibilities open for matching these vectors. We have termed them *quantitative* and *qualitative* methods. Quantitative refers to matching techniques based on the vectors themselves, such as size, orientation, and location. The likelihood of a correct match should be decided by the total number of vectors with identical quantities. Qualitative matching, on the other hand, is more similar to fingerprinting methods [3, 4] in that it is the connectedness of each vector which is compared. Points where convergenges/divergences, islands, or bifurcations exist are recorded from vector information, and these can be used as comparison points [4].

6.2 Pattern characteristics

The subset of animals mentioned for which pattern matching may be relevant includes cheetah, coelacanth, and giraffe. These animals have patterns characterised by spots or patches rather than stripes (Figure 7). PEMI has been designed to provide support for matching these designs in the future. In this case a pre-determined number of markings will be found automatically within a ROI, and recorded as point co-ordinates. It is these points which will be used for comparison rather than vectors. This method is related closely to those used in fingerprint techniques and the qualitative method described above, although every point will be of the same type.

Figure 7: The neck patches on a giraffe can be used for point matching.

Combining these matching techniques will provide support for analysing different patterns across various species, but not all. It is important to allow room for additional methods to be added to the system in future work, a factor which has been kept in mind while developing PEMI.

7 Conclusions

Wild animals with unique markings can be identified from photographic data. It is beneficial to automate this process, as the work is tedious and time consuming. ImageJ is a suitable basis for this analysis, since it provides routine housekeeping code and image processing tools.

Digital photographs may be unclear or contain unwanted artefacts, so the first step toward identification is preparation of the image itself. Common filters such as sharpen and smooth are effective in removing noise or enhancing edges. De-interlacing is often required if frames are captured from a video source. Binarisation by adaptive thresholding is suitable for image analysis as unwanted lighting and colour effects can be removed.

A region of interest is defined by the user. This provides flexibility in terms of alignment. Iterative sequential thinning is performed to produce a homotopic skeleton. This study has laid the foundations for information extraction and generalised pattern matching.

8 Outstanding Work

The image preparation stage is complete. However, the following steps are still outstanding: contour tracing of the homotopic skeleton; alignment of the vector co-ordinates; quantitative and qualitative matching; statistical reporting of findings.

9 Acknowledgements

This study was supported by the JRC fund at Rhodes University. Photographic data of zebra and giraffe were obtained at the Amakhala Game Reserve [9].

10 References

[1] Kelly, M.J. *Computer-Aided Photograph Matching in Studies Using Individual Identification: An Example from Serengeti Cheetahs.* Journal of Mammalogy, 82(2):440-449, 2001.

[2] Peterson, J.C. *An identification system for zebra.* E. Afr. Wild. J., 10:59-63, 1972.

[3] Roddy, A.R., Stosz, J.D. *Fingerprint Feature Processing Techniques and Poroscopy.* Intelligent Biometric Techniques in Fingerprint and Face Recognition, 37-105, CRC Press Intl. Series on Comp. Intell. 1999.

[4] Maio, D., Maltoni, D. *Minutiae Extraction and Filtering from Grayscale Images.* Intelligent Biometric Techniques in Fingerprint and Face Recognition, 155-192, CRC Press Intl. Series on Comp. Intell. 1999.

[5] Castleman, K.R., *Image Analysis: quantitative interpretation of chromosome images.* Image Processing and Analysis, A Practical Approach, Chapter 2. Oxford University Press, March 2000.

[6] Bernard, T.M., Manzanera, A. *Improved Low Complexity Fully Parallel Thinning Algorithm.* Proceedings 10th International Conference on Image Analysis and Processing (ICIAP'99), Venice, Italy, September 1999.

[7] ImageJ homepage: http://rsb.info.nih.gov/ij/
Author: Wayne Rasband, wayne@codon.nih.gov

[8] Sonka, M., Vaclav, H., Boyle, R. *Image Processing, Analysis, and Machine Vision.* 2nd Ed. 1999, Brooks/Cole Publ. Co.

[9] Amakhala game reserve. Visit: http://www.amakhala.com

[10] Foster, J.B., *The giraffe of Nairobi National Park.* E. Afr. Wild. J., 4:139-148, 1966.

[11] Douglas-Hamilton, I. *On the ecology and behaviour of the Lake Manyara Elephants.* E. Afr. Wild. J., 11:401-403, 1973.

[12] Barker, N., Rhodes University. Pers. comm.