Classification of Motion Behaviour of Animals using Supervised Learning Algorithms

Bachelor's Project Thesis

René Flohil, s2548925, r.t.flohil@student.rug.nl

Supervisors: Emmanuel Okafor, Porntiwa Pawara, dr. Marco Wiering

Abstract: Recognition of the world around us becomes more and more important in both entertainment and practical fields, the interest for research into recognition algorithms also has increased. Few studies have investigated the classification of behaviours of a given animal using machine learning algorithms. This thesis attempts to describe and compare the performance of two different feature detectors: Histogram of Oriented Gradients (HOG) and Image Pixel Intensity (IMG), and two different machine learning algorithms: a Support Vector Machine (SVM) and a Multi-Layer Perceptron (MLP) for recognizing the motion behaviours of goats. The results show that the algorithm IMG + MLP yields better performances than using a combination of HOG + SVM on a smaller train set. This indicates that raw intensity information matters more than using a HOG representation. However, on smaller test samples, all of the algorithms performed exceptionally well attaining a near perfect and similar performance level. The use of HOG + MLP yield better performance than IMG + MLP on a more diverse test set.

1 Introduction

Over the past years the fields of computer vision and automated recognition have gained increased popularity. Technologies like face detection have been around since 1994 (Yang; Huang, 1994) and are becoming more and more integrated into our daily lives. Image recognition is the process of understanding of what we see and what is happening around us (Shapiro, 1992). However as recognition, detection and classification of objects and animals are becoming well known areas for the world of computer vision, the recognition of what state these objects are in is still largely unexplored.

Questions such as 'what is this object?' and especially 'where is the object?' are not tough riddles to answer as many existing algorithms already solve these problems in very short computational times (Lowe, 1999) with great accuracies (Ren; Li, 2016). However the question 'What is this object doing?' is a question that has not been answered a lot. Studies have been done on behaviour recognition in insects (Noldus; Spink; Tegelenbosch, 2002) or the recognition of crowd behaviour (Cupillard; Bremond; Thonnat, 2003), yet both have focused on interaction or group size and not on individual behaviour.

The goal of this research is to study the progress of current technology in behavioural classification of animals using supervised learning techniques. The research question is stated as 'What combination of classical descriptor and classification model works best in recognizing animal behaviour?'

A dataset was needed in which one kind of animal performed multiple behaviours to use in a multi-classification problem. Most research in the field of computer vision carried out on animal datasets involves the development of recognition or detection systems. This thesis focuses on the use of feature descriptors each combined individually with different supervised learning algorithms. To achieve the stated aim a dataset was collected. The used dataset contains individual instances of 10 behaviours of goats.

This project compares two different classical feature descriptors in combination with two different supervised machine learning algorithms. The first feature detector used in the research is the Histogram of Oriented Gradients (HOG) which has become an increasingly popular feature descriptor

for detection problems that is described in Dalal and Triggs' paper (2005). HOG describes the distribution of normalized horizontal and vertical gradients in an image, which makes it useful for detecting edges and also contrast on objects. The HOG algorithm then transforms this information into a histogram based on the respective orientations. This research investigates whether the used feature descriptor combined with supervised learning algorithms can be considered as a good recognition system to deal with the above stated problem. The second feature descriptor that will be used is the Raw Pixel Intensity (IMG) feature which is described as a global feature descriptor as it converts the image into a histogram without altering of any pixel information.

The histograms that are produced are fed as a feature abstraction to either a Support Vector Machine (SVM), often used for two-group classification problems (Cortes; Vapnik, 1995), or a Multi-Layer Perceptron (MLP), which has long been used as a classifier (Rumelhart; Hinton; Williams, 1986). This thesis explains the acquisition of an own dataset. That is done through collecting videos online, extraction of sequential video frames, cropping out the region of interest (RoI) containing the presence of goat(s). This is done to minimize irrelevant information and finally separating the images into their respective classes. The workings of the classical feature descriptors and a small overview of the workings of the SVM and MLP are described herein. Finally a comparison and discussion of the results obtained are given.

2 Method

2.1 Dataset Collection

To perform the research a dataset is needed that contains enough classes of different motion behaviours of goats. To create this dataset video footage will have to be gathered. The algorithms will use the data of sequential video frames of these pictures, it is thus important that there is enough video footage. For a video that has a frame-rate of 60 frames per second, a video that contains 5 seconds of clear vision of the goat is enough as that will provide approximately 300 images. In theory 10 videos exhibiting different motion behaviours will

be enough to fill the dataset.

Once enough videos are collected to satisfy the amount of classes that is deemed enough it is important to crop out regions of interest (RoI) of the image content and eliminate the redundant part of the image where no goat is present and make sure the behaviour of the goat is fully exhibited. For example, if one wishes to classify a flocking behaviour, one would need to include multiple goats in the image and simultaneously exclude as much background as possible.

Before this is done however the videos need to be split into the sequential video-frames that make up the video, such that the set of videos V consists of the video frames that make up the videos:

$$V_n(Q) = \sum_{n=1}^{N} f_Q(n) \qquad (2.1)$$

where Q denotes the number of frames for a given video V and f_Q is the amount of frames in class n. Please note that Q varies depending on the streaming duration of each video V_n. This means that there exists a non-uniform number of frames per video.

After the sequential video frames have been extracted and the images were cropped to remove unimportant information the frames are put into N different classes. The dataset consists of several behaviours exhibited by domesticated goats (Hansen, 2015) as well as wild goats (Miranda-de La Lama; Mattiello, 2010); butting, eating, fainting, flocking, mounting, resting, running, pooping, sleeping and standing. The used dataset contains a total of 3588 images and consists of ten different classes. Some examples of the dataset are shown in Figure 2.1.

2.2 Dataset Partitioning

The dataset earlier discussed is partitioned into several entities. The classification algorithms that will be used have their own variable parameters: the C-parameter for the Support Vector Machine and the amount of nodes in the hidden layer of the Multi-Layer Perceptron. Both are tuned to obtain the best possible recognition system. This is achieved by using two distinct dataset distributions; the first distribution (80%-20%) and the second distribtuion (50%-50%) for the (training and testing sets) respectively. This means that the
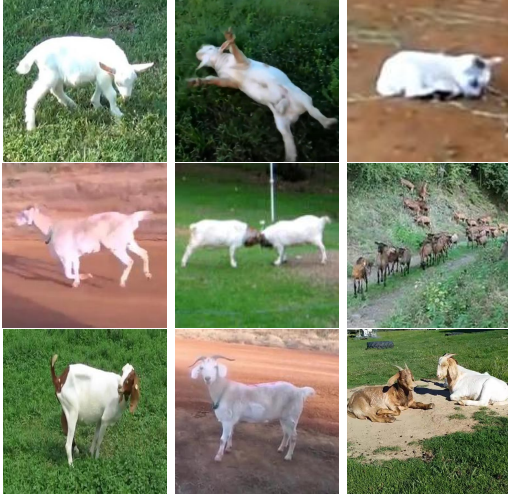
Figure 2.1: Individual instances of goat behaviours from the used dataset.

first dataset distribution is partitioned into test, validation and training sets in the ratio 20%, 10% and 70% respectively. The train-validation splits were repeated for 5-fold cross-validation.

Moreover another set of experiments was examined to investigate the classification accuracies on two different sets of splits (50%-50%) in the second dataset distribution. For more clarity the first distribution test set can be referred to as Test 1, while the second test set is called Test 2. Tests done on a more diverse test set containing 989 images are referred to as Test 3. More information on Test 3 is given in Section 3.3.

2.3 Feature Descriptors

After the dataset has been collected and split into the earlier mentioned distributions the images are fed into two separate feature descriptors; the local feature descriptor (HOG) and the global feature descriptor (IMG). The feature descriptors that are used for this study are discussed now.

2.3.1 Histogram of Oriented Gradients (HOG)

The HOG is computed by creating $n \times n$ patch blocks from a given image. Then the effective magnitude gradients of each patch block with respect

to their orientation bins are calculated to produce a feature vector of a particular image. The image is then divided in 8×8 cells and a histogram is created for each of these cells. Figure 2.2 shows the gradients point towards the direction of change in pixel intensity. The size of the arrow correlates with the intensity of the change. These gradients are then normalized per 16×16 block. The magnitude of gradients is calculated using the following;

$$M_G = \sqrt{G_x + G_y} \qquad (2.2)$$

The orientation of the gradient as follows;

$$\theta = tan^{-1}\left(\frac{G_y}{G_x}\right) \qquad (2.3)$$

where M_G denotes the magnitude of the gradient, G_x and G_y denote the horizontal and vertical gradients and θ denotes the orientation of the gradient.
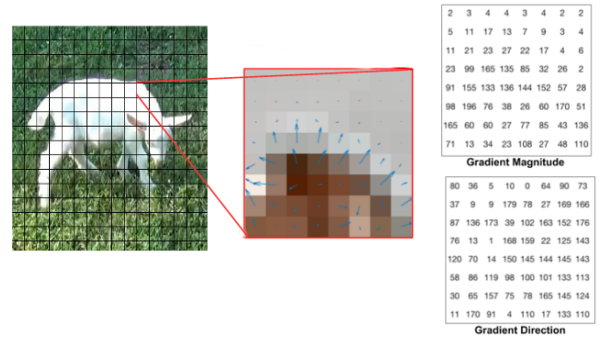

Figure 2.2: Example of how HOG is divided into cells. The calculated gradients point towards the largest change in pixel intensity.

2.3.2 Raw Pixel Intensity (IMG)

The IMG feature descriptor is a lot simpler. It simply converts the image data into a greyscale version of the image which in turn is converted into a histogram. The bins are computed as the product of the image resolution $200*150 = 30.000$ based on the gray level intensities. Then the supervised learning algorithms use the information to construct classification models. Because it doesn't extract any local features like HOG does, IMG is also called a global feature descriptor. A simple illustration can be found in Figure 2.3.

The histograms that are created by the feature descriptors will then be used for the training and construction of the classification model. The effectiveness of the classification model is measured on an unknown test set in both Test 1 and Test 2.
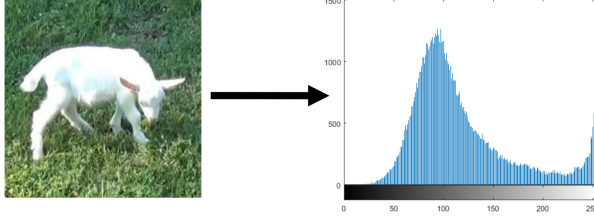

Figure 2.3: Example of how IMG uses the values of the input images as a histogram. The IMG descriptor uses 30.000 feature dimensions for each image.

2.4 Classification Algorithms

The histograms that are created will be fed into two different classification algorithms to evaluate the performance of these algorithms. It is important to note that all possible combinations of one feature descriptor with each of the supervised learning techniques will be evaluated. The supervised learning algorithms used for this study are discussed as follows;

2.4.1 Support Vector Machines (SVM)

One classification model that will be used is the Support Vector Machine (SVM) (Cortes et al., 1995). The algorithm works by placing these vectors in a feature space as it attempts to create a dividing margin between two different classes and then maximizing that margin.
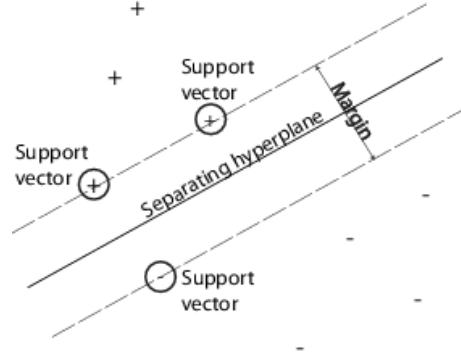

Figure 2.4: Illustration of a Support Vector Machine. (Support Vector Machines for Binary Classification, n.d.)

Figure 2.4 shows an illustration of a Support Vector Machine. The circled support vectors, which are on the edge of their 'class space' are used to maximize this margin. For a linear multi-class SVM, the output $z_k(x)$ of the k-th class can be computed as:

$$z_k(x) = w_k^T i(x) + b_k \qquad (2.4)$$

In this research $i(x)$ are the input vectors which are created by either the HOG or IMG feature descriptors from image x. The linear classifier for class k is trained to output a weight factor w_k with a bias value b_k (Okafor; Pawara; Karaaba; Surinta; Codreanu; Schomaker; Wiering, 2016).
Two different loss functions are used. The first loss function method is called the L1-SVM. The second loss function is the L2-SVM. The L2-SVM classifier is defined as:

$$\min_w \frac{1}{2} w^T w + C \sum_{i=1}^{n} (\max(0, 1 - y_i z_k(x)))^2 \quad (2.5)$$

(Fan; Chang; Hsieh; Wang; Lin, 2008).
Here $y_i = \{1, -1\}$ where $y_i = 1$ if x_i belongs to the k-th classifier and $y_i = -1$ if x_i does not belong to the target class. C is the penalty parameter. When the SVM is trained the margin is determined. The prediction of the class label of new instances is done by checking where the new vector is positioned in the feature space and on what side of the margin it is present. In other words the classifier (Tang, 2013) then outputs predicted class labels to an image x using:

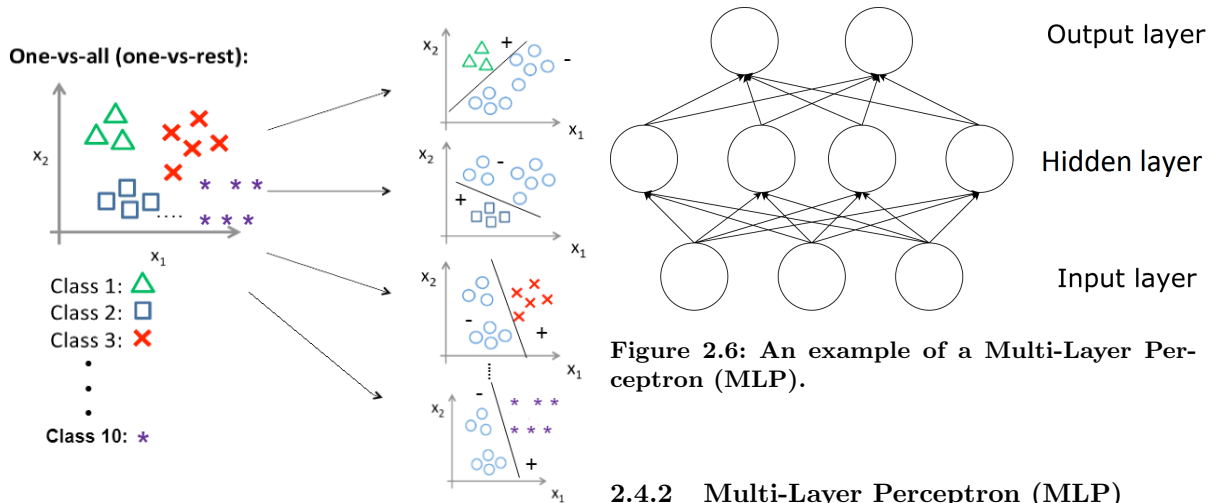$$arg \max_k (z_k(x)) \qquad (2.6)$$

Figure 2.6: An example of a Multi-Layer Perceptron (MLP).

Figure 2.5: Illustration of a multi-classification problem (1-vs-All) solved using a Support Vector Machine. Adapted from (Ng, 2018).

2.4.2 Multi-Layer Perceptron (MLP)

The second classification model that will be used in this research is the Multi-Layer Perceptron (MLP) (Rumelhart et al., 1986). An MLP works by using three different kinds of layers that consists of nodes (as shown in Figure 2.6). All nodes between two layers are connected. The first layer is the input layer in which the feature vectors are fed. In this research this means that the feature vectors produced by HOG or IMG are provided to the input layer or the hidden layer.

Second, the hidden layer consists of a variable amount of nodes. It is here that the weights of the nodes are altered as to train the MLP. These weights are altered using a forward-backward propagation. This research uses a scaled conjugate gradient backpropagation for training the loss-function. We used a cross-entropy loss function. The hidden layer uses a hyperbolic tangent sigmoid activation function to compute feature activations. Finally, the output layer represents the performed classification. In this research the output layer consists of 10 nodes, each representing one of the 10 different classes. The output layer uses a softmax activation function.

However the workings of the SVM provide a problem for our classification problem. As can be seen in Figure 2.4 the Support Vector Machine solves a 1-vs-1 classification problem, but our dataset contains ten different motion behaviours and thus ten different classes. We thus need to solve a multi-classification problem with a 1-vs-1 classifier. Fortunately the solution is a simple one. SVMs are binary classifiers, but can be extended for multi-classification (Lingras; Butz, 2007). The multi-classification problem is a ten-fold 1-vs-All classification problem where one class is labeled as positive and all other classes are labeled as negative as is illustrated in Figure 2.5. This is then done ten different times, one time for each class.

The SVM uses a C-parameter to influence the SVM optimization. High C-values cause a smaller margin-hyperplane if that yields a higher accuracy. Lower C-values cause a larger margin-hyperplane to be created even if that causes a drop in accuracy. To achieve optimal accuracies for all SVM classification models a parameter tuning is done on the second data distribution. The parameter that yields the highest accuracy is chosen for the evaluation of the classification algorithm.
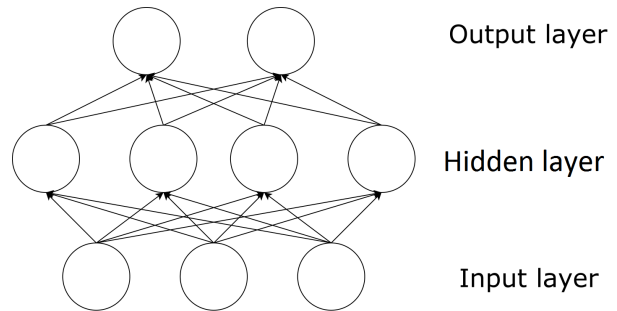
Furthermore the algorithm was trained for a maximum of 3000 epochs in the case that the gradient method does not stop the learning phase of the algorithm. Similar to the SVM a parameter tuning is done on the second data distribution, but with the amount of nodes in the hidden layer instead of the C-parameter. Here too the value that yields the highest accuracy is chosen for the evaluation of the classification model.

3 Results and Discussion

3.1 Determination of the best hyperparameters for the supervised learning algorithms

Before the evaluation of the classical descriptors and the classification algorithms can happen the hyperparameter for the SVM and the amount of nodes for the MLP need to be determined.

3.1.1 Determination of the SVM's C-parameter

The hyperparameter for the SVM is the C-parameter. The choice of an optimal C-parameter was determined by carrying out a grid search in the bounds $[-5 \leq x \leq 5]$ over an interval of 1. The C-parameter uses 2^x resulting in a bound within the range $[\frac{1}{32} \leq C \leq 32]$. An exception was made for the IMG + L2-SVM method where a grid search was done in the bounds $[\frac{1}{32} \leq C \leq 1024]$ as shown in Figure 3.1. We observed that the fractional values of the exponent as presented in Table 3.1 yielded the best results.

In Figure 3.2 the parameter tuning results of the Support Vector Machine in the training phase are shown. The parameters that were used in the evaluation phase of the SVM can be seen in Table 3.1. Figure 3.3 shows the test results of the parameter tuning, however only the results of the training phase were used for the determination of the parameters. The train accuracies in the aforementioned figure show that peak performances are attained after C exceeds 4, a similar result is seen in the test accuracies with the exception of the IMG + L1-SVM which attains a peak performance of approximately 99% and the IMG + L2-SVM which attains a peak performance of 94%.

3.1.2 Determining the amount of nodes in the MLP's hidden layer

On the other hand the number of nodes in the hidden layer (N_H) of the MLP is tuned using the bound $[10 \leq N_H \leq 210]$. In Figure 3.4 the results for the MLP are shown. The parameters that were used in the evaluation phase can be seen in Table 3.2. The MLP attained near perfect performances

up to approximately 96.7% for the IMG feature descriptor. Thus MLP with a hidden layer size of 70 nodes was picked for the method (IMG + MLP) and a layer with 110 nodes for the (HOG + MLP) method.

Method	C-parameter
HOG + L1-SVM	5.66
HOG + L2-SVM	16.00
IMG + L1-SVM	11.31
IMG + L2-SVM	90.51

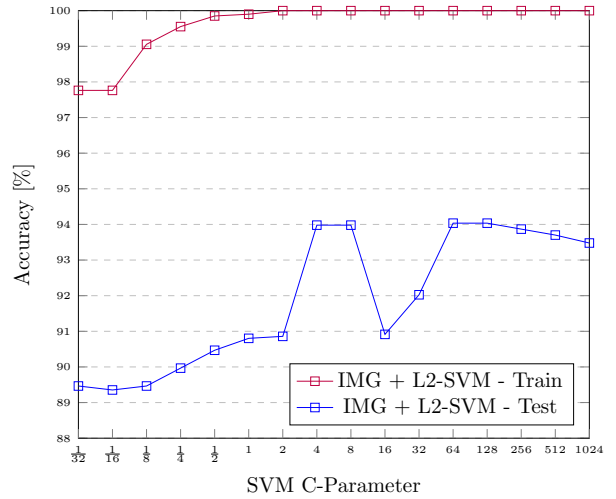**Table 3.1: The best-found C-parameters for the SVM for a given feature descriptor.**

Figure 3.1: Train and test accuracies using IMG + L2-SVM for classifying goat behaviours using an SVM with a C-value in $[\frac{1}{32} \leq C \leq 1024]$.

Method	No. of nodes
HOG + MLP	110
IMG + MLP	70

Table 3.2: The best-found number of nodes for the MLP for a given feature descriptor.

3.2 Cross-Validation & Evaluation

The evaluation of the classification models is based on a five-fold cross validation. Additionally we examined the performance of the classification models on two test sets; Test 1 and Test 2. The summary
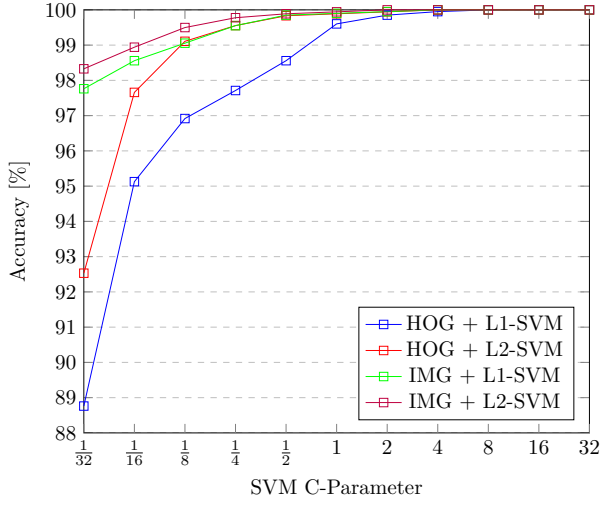
Figure 3.2: Training accuracies using L_n-SVM combined with classical descriptors for classifying goat behaviours using an SVM with a C-value in bound $[\frac{1}{32} \leq C \leq 32]$.

Figure 3.4: Test evaluation of MLP combined with classical feature descriptors for classifying goat behaviours by varying the amount of hidden layer nodes in the MLP using the bound $[10 \leq N_H \leq 210]$.

Test 2. However the HOG + MLP and IMG + MLP methods outperform all other methods in both Test 1 (perfect accuracies) and Test 2 with accuracies of 95.48% and 95.68%. Based on the performances of the MLP classifier on the two feature descriptors as shown in Table 3.3 we only considered the MLP for the remaining experiments.
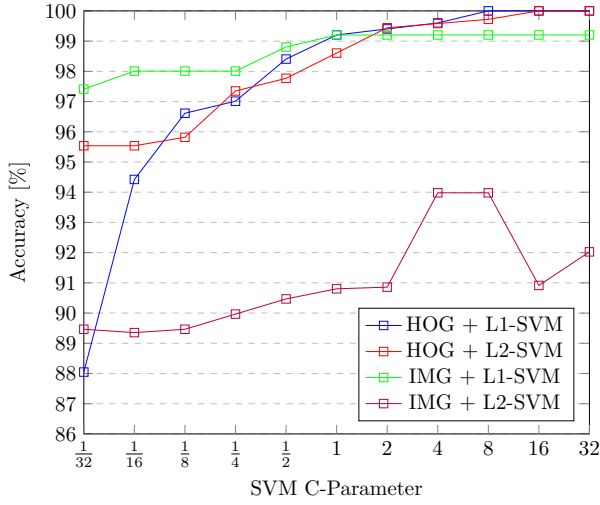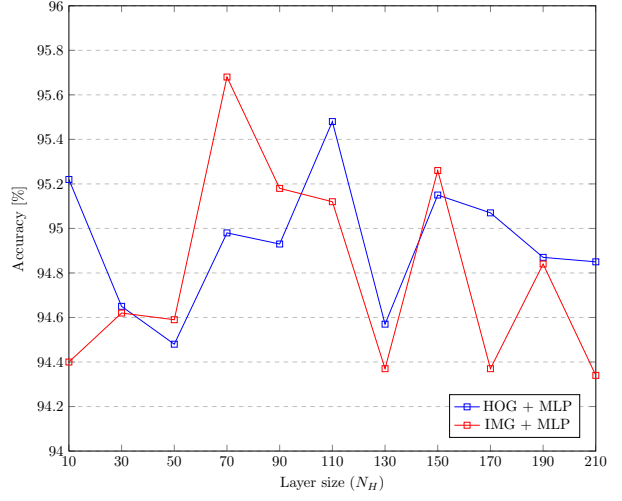

Figure 3.3: Test accuracies using L_n-SVM combined with classical descriptors for classifying goat behaviours using an SVM with a C-value in bound $[\frac{1}{32} \leq C \leq 32]$.

Methods	Validation	Test 1	Test 2
HOG + L1-SVM	99.97 ± 0.08	100%	89.02%
HOG + L2-SVM	100.00 ± 0.00	100%	89.02%
IMG + L1-SVM	99.66 ± 0.17	99.94%	90.91%
IMG + L2-SVM	99.79 ± 0.20	99.86%	95.48%
HOG + MLP	100.00 ± 0.00	100%	95.48%
IMG + MLP	99.98 ± 0.02	100%	95.68%

Table 3.3: Test performance of the recognition systems for two test distributions based on the best-found C-parameter described in Table 3.1, and hidden layer size.

of the results are reported in Table 3.3.

The second column from the mentioned table shows that all methods attain near perfect accuracies in the five-fold cross validation. Both the IMG + L1-SVM and IMG + L2-SVM methods outperform the HOG + L1-SVM and HOG + L2-SVM methods in

3.3 Evaluation on a unique dataset

The results shown in Table 3.3 suggest that all algorithms perform exceptionally well in recognizing the images of motion behaviours of goats. The discussion in section 3.4.1 poses that this is a flawed
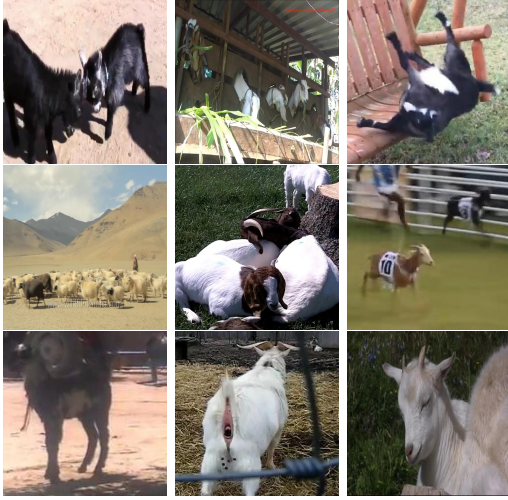
Figure 3.5: Some example images of the unique dataset describing behaviours of goat. Behaviours from top left to bottom right: butting, eating, fainting, flocking, resting, running, standing, pooping and sleeping.

Method	Test 3	Test 3
HOG + MLP	81.94 ± 0.94	83.04 ± 1.28
IMG + MLP	81.73 ± 0.41	82.03 ± 0.68

Table 3.4: Test performance of the recognition systems for two test distributions using HOG + MLP (No. of nodes = 110) and IMG + MLP (No. of nodes = 70) on Test 3. The second and third columns describe performance using 80% and 50% train set distributions of the original dataset respectively.

3.4 Discussion

This research has demonstrated the capabilities of the two classical feature descriptors: HOG and IMG, and two classification algorithms: the Support Vector Machine and the Multi-Layer Perceptron in recognizing motion behaviour in animals, or goats to be more specific. The question remains: what combination of classical descriptor and classification model works best in recognizing animal behaviour?

3.4.1 Implications

Based on the examined dastaset the MLP methods outperform the other methods in Test 2. Another observation is that the IMG representation combined with SVM variants yield performances that surpass results obtained from HOG combined with SVM. This goes against the original hypothesis which states that 'the algorithms augmented with HOG would perform better as simplifying images would open up the possibility to recognize a broader spectrum of images'. To account for the variations in one specific motion behaviour in a given image it was expected that the extra information that remained in the histograms that are produced by the IMG feature descriptor would overfit the data and thus it would perform worse on new examples. However the IMG methods outperformed the HOG methods in Test 2. Furthermore the overall precision of all methods was almost perfect on a small test set in Test 1.

To find out why this may be we have to take a look at the dataset and the way it was collected. As described in section 2.1 the frames that are extracted from the videos are sequential. These frames are taken from sections of videos that are approxi-

notion.

To provide another perspective on the classification system, a new dataset was collected containing the same classes, but with unique images such that there are no identical images between the new dataset and the original dataset. The unique dataset contains a total of 989 images and a sample of the dataset can be seen in Figure 3.5. Tests done using this dataset as test set are referred to as Test 3.

The training set distributions from the original dataset i.e. 80% and 50% were used when training the MLP classification models, which is finally evaluated on Test 3. The results obtained are reported in Table 3.4. The performances reported in the table were based on five repeated runs. As discussed earlier the results in Table 3.3 show that MLP performed equal or better than all SVM methods on the original dataset. Thus the tests performed on the unique dataset were done using HOG + MLP and IMG + MLP. Table 3.4 shows that both recognition systems perform worse on unique test sets. (HOG + MLP) reaches accuracies of 83% and (IMG + MLP) reaches accuracies of 82% on runs using a 50% train set distribution.

mately 5 seconds long with most samples for one class taken only from one video. This implies that all images in one class are extremely similar, the intraclass differences are small, as most classes only contain samples of one video. The result is that the images in the testing phase are very similar, yet not identical to the images in the training phase. The small intraclass difference might cause an overfitting of each class meaning that new images that do fit into one of the classes, for example another image of two goats butting, but in a different setting, may not be classified correctly as the lower accuracies of Test 3 show us.

Another reason may be that the interclass differences are large, this is due to the fact that the images that make up the classes are all extracted from different videos for each class. This also explains why IMG performs better than HOG in Test 1 and 2, even though a lot of extra information is removed from the images by cropping out the region of interest where the goat is. The colour of the goat or the small snippets of background color may be enough to recognize the difference between two classes, not because of different behaviours, but because of different (background) colors. Not only are the behaviours of the goats in two classes different (the criteria on which it is desired that the classification models differentiate between the classes), the whole environment of the goats are different. This includes colour of the goat, colour of the background and objects in the background. One could suggest that if the trained models were subjected to a goat that performs a certain behaviour that can been found in class A, but with a background that can be found in class B the model would not classify the image correctly. The results of Test 3 show us that this is a factor as the accuracies attained are lower than that of Test 1 and 2. However the factor is not huge as accuracies of approximately 82% are still decent. We can conclude that the cause of the high accuracies is partly due to a very simple dataset.

3.4.2 Improvements

To increase the 'difficulty' of the dataset the intraclass and interclass differences need to be fixed. A crucial part of visual classification systems is being robust to intraclass differences (Gehler; Nowozin, 2009) yet the dataset's differences are too small.

These intraclass differences can be made larger by using non-sequential frames of videos. This way the same video can be used, but the similarity between two images will be larger increasing the feature space of the class and ensuring that new images will be properly classified.

Another solution would be to include more videos into one class. If the frames that make up a class are extracted from multiple videos, but still show goats exhibiting the same behaviour not only would this increase the intraclass differences, it would also decrease the interclass differences and be a more broad depiction of a certain behaviour as the background will not hold any key information to the behaviour, instead only the goat would.

Another way is to make sure that all classes are collected from the same video. By making sure that all behaviours are collected in the same environment the background information would no longer be crucial to classifying the behaviour. This can be done by capturing the footage yourself instead of collecting the videos online as was done in this research.

4 Conclusions

In this research we have tried to compare the performances of a Multi-Layer Perceptron and a Support Vector Machine combined with either a Histogram of Oriented Gradients or Raw Pixel Intensity (IMG) feature descriptor as classification models for recognizing motion behaviours of goats in a still image.

Based on a five-fold cross validation and the performance using a 50% and an 80% train set distribution the use of MLP works best in a 50%-50% dataset partition. In this same dataset partition the use of SVM combined with an IMG feature descriptor outperforms an SVM combined with a HOG feature descriptor. Furthermore almost all methods' training performance are approximately 100%. We demonstrated that these classification models were robust in recognizing motion behaviours in still images and in a more practical problem the recognition system is robust using both HOG + MLP and IMG + MLP on a test with diverse image samples.

This research has thus demonstrated the use of machine learning to predict animal behaviours and/or

motion dynamics with a scalable dataset encompassing a diversity of several videos per class under varying environmental conditions.

It will be interesting to investigate the use of Convolutional Neural Networks (CNN) compared to the examined methods for the recognition of animal behaviours. This is due to the success of CNN in several classification challenges such as animal recognition (Okafor et al., 2016).

References

[1] Yang, G., Huang, T. S. (1994). Human face detection in a complex background. *Pattern recognition, 27*(1), 53-63.

[2] Shapiro, S. C. (1992). Encyclopedia of artificial intelligence, second edition. *John*

[3] Lowe, D. G. (1999). Object recognition from local scale-invariant features. In Computer vision, 1999. *The proceedings of the seventh IEEE international conference* on (Vol. 2, pp. 1150-1157). IEEE.

[4] Ren, H.; Li, Z. N. (2016) Object detection using boosted local binaries. *Pattern Recognition*, 60, 793-801

[5] Noldus, L. P., Spink, A. J., Tegelenbosch, R. A. (2002). Computerised video tracking, movement analysis and behaviour recognition in insects. *Computers and Electronics in agriculture*, 35(2-3), 201-227.

[6] Dalal, N.; Triggs, B. (2005). Histograms of oriented gradients for human detection. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference*, 1, 886-893.

[7] Cortes, C., Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3), 273-297.

[8] Rumelhart, D. E., Hinton, G. E., Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088), 533.

[9] Okafor, E., Pawara, P., Karaaba, F., Surinta, O., Codreanu, V., Schomaker, L., Wiering, M. (2016, December). Comparative study between deep learning and bag of visual words for wild-animal recognition. In *Computational Intelligence (SSCI), 2016 IEEE Symposium Series* on (pp. 1-8). IEEE.

[10] Mathworks Documentation. (n.d.). Support Vector Machines for Binary Classification. Retrieved from https://nl.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html

[11] Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008), Liblinear: A library for large linear classification, *The Journal of Machine Learning Research,* vol. 9, pp. 18711874.

[12] Tang, Y., (2013). Deep learning using linear support vector machines, in *Challenges in Representational learning, The ICML 2013 Workshop on,*

[13] Ng, A. (2018) Coursera Machine Learning Course. Retrieved from https://www.coursera.org/learn/machine-learning, https://www.youtube.com/watch?v=vNNcFTd_630

[14] Lingras, P., Butz, C. (2007). Rough set based 1-v-1 and 1-v-r approaches to support vector machine multi-classification. *Information Sciences*, 177(18), 3782-3798.

[15] Hansen, I. (2015). Behavioural indicators of sheep and goat welfare in organic and conventional Norwegian farms. *Acta Agriculturae Scandinavica, Section AAnimal Science*, 65(1), 55-61.

[16] Miranda-de La Lama, G. C., Mattiello, S. (2010). The importance of social behaviour for goat welfare in livestock farming. *Small Ruminant Research*, 90(1), 1-10.

[17] Andersen, I. L., Be, K. E. (2007). Resting pattern and social interactions in goats the impact of size and organisation of lying space. *Applied Animal Behaviour Science*, 108(1), 89-103.

[18] Gehler, P., Nowozin, S. (2009, September). On feature combination for multiclass object classification. *Computer Vision, 2009 IEEE 12th International Conference* on pp. 221-228. IEEE.

[19] Knuth, D. (1984). *Computers and Typesetting*, Retrieved from http://www-cs-faculty.stanford.edu/knuth/abcde.html