Automatic Annotation of Social Behavior of Voles

Aakriti Lakshmanan

Under the direction of

Dr. Shuyu Wang and Ms. Kara Quine
Manoli Laboratory
University of California at San Francisco

Research Science Institute

Abstract

The study of social attachment behaviors and the factors behind them is integral for developing a deeper understanding of many neuropsychiatric conditions with social deficits, such as autism and schizophrenia. The main animal model used in these studies is *Microtus ochrogaster*, commonly known as the prairie vole, due to its tendency to form lifelong socially monogamous pair bonds. However, studies of naturalistic social behaviors require time- and labor-intensive manual annotation, which is also limited by imperfect inter-rater reliability. Therefore, developing a system for automatic annotation of these behaviors can make these studies more accessible. This study aimed to develop a set of features from computationally defined vole contours, and also build a classification model for huddling, an attachment behavior specific to prairie voles. The average F1 scores for the presence and absence of the behavior were to 97% and 98%, respectively, with an accuracy of 98%. The features that were deemed to be most important for the huddling classifier were the area, minor axis length and eccentricity of the center vole. Future directions for this study are focused on optimizing the current model and developing classifiers for other vole behaviors. Developing a classification model for huddling behaviors in voles is an important step in increasing the accessibility and accuracy of social behavior studies.

Summary

Studying social attachment is important because deficits in social interactions are a hallmark feature in many mental disorders, including autism and schizophrenia. Prairie voles have emerged as the premier system for studying social attachment because they form lifelong socially monogamous pair bonds both in the wild and in the laboratory. However, studies of prairie vole behavior are highly labor-intensive due to the need for manual behavior annotation. Moreover, individual biases in manual annotation could render attempts to compare behavior label sets less meaningful. Therefore, creating an automatic system to mark the existence of behaviors can help with this problem and make studies of this type more accessible. A classification model was created using a machine learning model known as the random forest. The overall accuracy of the final model was 98%. The most important features were the area, minor axis length, and roundness of the center vole. Future directions for this study are focused on improving the current model and developing classifiers for other vole behaviors. Developing a classification model for huddling behaviors in voles is an important step in increasing the accessibility and accuracy of social behavior studies.

1 Introduction

Social attachment behaviors are an important part of human development. They are exhibited in humans in many ways, from friendships and familial attachment to relationships. Many conditions such as schizophrenia and autism spectrum disorders are characterized by impaired social attachment behaviors. Therefore, analyzing the neurobiological factors behind these behaviors is necessary to learn more about these conditions and how to treat them. [1]

Animal models have been an essential part of comparative psychology studies for many years. Animal models also allow scientists to integrate neural activity and behavior, which is important for understanding the underlying causes of specific behaviors. However, common animal models such as mice and flies do not exhibit social attachment behaviors. While mice can be colonial in the wild, these behaviors do not translate into an artificial laboratory setting.[2] [3]. The lack of social behaviors in common animal models can make it difficult for scientists to study these behaviors.

By contrast, prairie voles, otherwise known as Microtus ochrogaster, exhibit robust social attachment behaviors.[4] Even in an artificial laboratory setting, most prairie voles are monogamous and will selectively bond with one animal for a lifetime which allows scientists to examine these behaviors in-depth [5]. Monogamous animals are bi-parental and reject intruders of the opposite sex. [6] Given inter-species conservation of the molecular substrates of social attachment, and potentially of cellular and circuit-based features as well, scientists have focused on examining the prairie vole model to gain insights into social attachment behavior selection.

A common assay for examining social attachment is known as the partner preference test, in which a focal vole is allowed to roam freely in a three-chamber apparatus and scored on the amount of time it spends with its bonded partner (tethered to one end chamber) versus an intruder vole (tethered to other end chamber), as seen in Figure 1 [5]. Behavior assays can last as long as three hours and can contain many different behaviors in rapid succession. Currently, the gold standard for behavior annotation requires reducing the speed of the video several fold to enable detection of subtle and rapidly evolving movements. In addition, due to the fact that videos are annotated by different individuals who may have undergone different training, individual biases can affect the accuracy and precision of results. The long turnaround of this process has limited many studies of this type. Therefore, automation of the behavior annotation process is integral for increasing accuracy and precision, as well as reducing the turnaround time [7]. Automation of this process may also uncover subtle

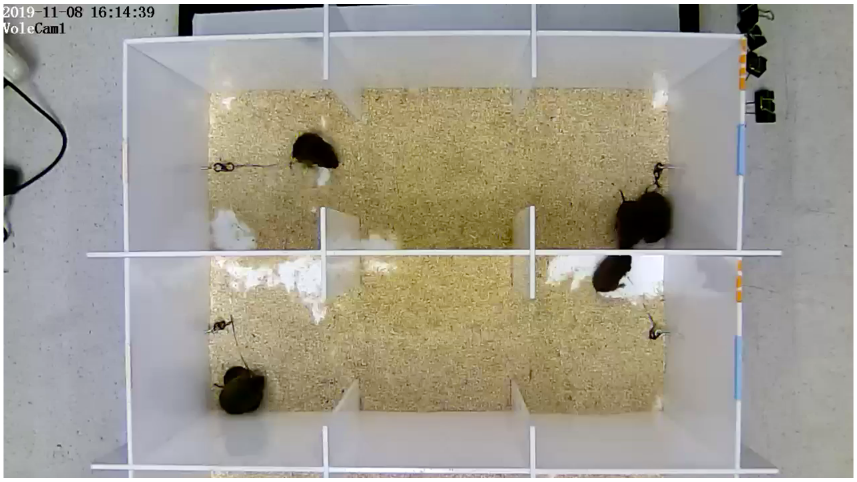

Figure 1: Two partner preference tests (top row and bottom row). Each row contains three different chambers with the left and right vole restricted to their respective chambers. The center vole is set to roam free. The separate chambers act as a way for researchers to determine any bias that the center vole may have towards another vole.

behaviors that were previously overlooked during manual annotation.

Multiple programs have already been created to automate behavioral annotation of animals. However, many of these programs have been created for the behaviors of other organisms, such as mice [8][9] and drosophila [10]. Since these organisms do not exhibit social behaviors like those seen in voles, pre-existing annotation systems do not have models that can be used to distinguish vole-specific social behaviors. In addition, some behaviors require analysis of multiple animals at once, which further reduces the applicability of platforms focused on one organism's singular behavior. Systems currently available for multiple animals utilize the color of animal coats to distinguish between animals. [9] However, as seen in Figure 1, the voles used in these assays are the same color, so these systems also are not applicable. Therefore, there is a need for an automated system for tracking the movements of multiple voles and their social behaviors.

This study aims to examine the relationships between vole body contours and behaviors. By definition, contours are a series of points that delineate the boundary of a region that exceeds a specified pixel intensity threshold. In my data, a contour can capture either a single vole or a pair of interacting voles. I focus on three main groups of parameters. Intrinsic parameters focus on the contour itself, such as shape. Kinematic parameters analyze the motion of the contour over frames, while location parameters examine the location of the contour in relation to other factors. Furthermore, this study aims to develop a classifier for

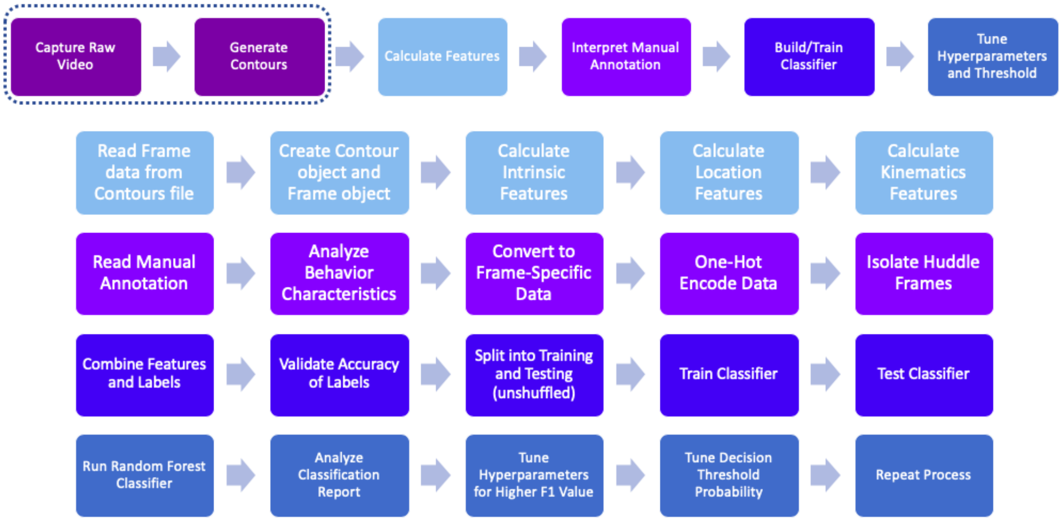
Capture Raw Video
Generate Contours
Calculate Features
Interpret Manual Annotation
Build/Train Classifier
Tune Hyperparameters and Threshold
Read Frame data from Contours file
Create Contour object and Frame object
Calculate Intrinsic Features
Calculate Location Features
Calculate Kinematics Features
Read Manual Annotation
Analyze Behavior Characteristics
Convert to Frame-Specific Data
One-Hot Encode Data
Isolate Huddle Frames
Combine Features and Labels
Validate Accuracy of Labels
Split into Training and Testing (unshuffled)
Train Classifier
Test Classifier
Run Random Forest Classifier
Analyze Classification Report
Tune Hyperparameters for Higher F1 Value
Tune Decision Threshold Probability
Repeat Process

Figure 2: Pipeline describing the separate steps in feature and classifier development. The initial steps, circled above, were completed previously.

a specific type of vole behavior known as huddling by using the calculated parameters as features for the model.

2 Methods

Overview

The data used to develop the following features and classifier were sourced from several videos of the partner preference assay taken at the Manoli Lab at UCSF. These videos ranged from 2 minutes to 1 hour in length and had a frame speed of 30 frames per second. To develop the full data-set, data from 12 videos were grouped for a total of 699883 frames. An overview of the entire process is shown in Figure 2. The videos had already been pre-processed to generate continuous contours for each video. To build a supervised classifier, it is necessary to develop labels and features. Features are descriptive values that act as input while labels are what the classifier is meant to predict, or in other words, the final output of the classifier.

Features

A set of features was created for each vole contour in each frame to correlate the individual contours to behaviors. 21 total features were created for each contour. Additional features were also created by averaging current features over either 5, 10 or 15 frames. Overall, the total number of features per frame was 138. The features can be split into three main groups: intrinsic, kinematics, and location.

Location Features

Location features were created to examine the location of each contour and the potential relationships that the location can have on the behavior exhibited in the frame. The centroid, computed using the moments of each contour, was used to approximate the location of each contour. Using this centroid, multiple other location parameters were created including the distance between voles and the distance from the corners of the chamber as shown in Figure 1. In addition, the distance between each point of the contour and the centroid was calculated. A low standard deviation of this set of data can signify a rounded or less eccentric contour.

Kinematic Features

How a contour moves over frames is integral to understanding what behavior is occurring as each behavior exhibits a certain style of movement. For example, in some behaviors, the vole pairs do not move frequently and remain in one location over time. Therefore, these features are important inputs for a classifier to distinguish between frames where a behavior is occurring and frames where there is no behavior occurring. Multiple different kinematics features were calculated, including velocity, acceleration, distance traveled, and direction of travel. Upon analysis of these parameters, it was shown that there were outliers in the data due to the obstruction of contours in specific frames. Therefore, a Savitzky–Golay filter was used on kinematics features to smooth out data and make them more reliable for use in a classifier. [11]

Intrinsic Features

While kinematic features look at the behavior of a contour over multiple frames, there are features in singular frames that might also aid in the annotation process. When two contours come into contact with one other, they merge into one singular contour. Therefore, looking at intrinsic features such as area and curvature can help predict such events. Much of the intrinsic features of a vole contour are highly dependent on which way the vole is facing. For example, features such as angle of orientation cannot be calculated for a full angle range without data signifying the direction the vole is facing. In order to get this data, dimensionality reduction was performed on the contours. This reduced the two-dimensional contour to two one-dimensional lines, otherwise known as the major and minor axis, which was done by applying principal component analysis on each contour. The resulting eigenvectors of the data were combined with centroid coordinates to calculate approximate lines for both the minor and major axis. Endpoints of each line were found on the outline of each contour to derive the length of each axis. Using these endpoints, features such as angle of orientation and eccentricity can be calculated. Eccentricity was defined as the ratio of the lengths of the major and minor axis, as a value closer to 1 represents a more circular contour. In addition, an ellipse was fitted to each contour to provide a smoothed representation of the original contour.

Labels

In order to get labels for use in a machine learning classifier, data from a set of manually annotated videos was used. This data was also sourced from the Manoli Lab at UCSF. These videos were manually annotated for the beginning and end of specific behaviors. In order to make this data practicable for use in a classifier, the data was one hot encoded, and a specific integer value was assigned to each behavior. While there are many different behaviors, as summarized in Figure 3, the manual annotation files used for the classifier grouped many of the sniffing behaviors together as an ”interaction behavior.” In addition, grooming, rearing, and attach behaviors were scarce. Due to data and time limitations, this study aims to look specifically at the huddling behavior and develop a classifier to predict this behavior in other videos.

Random Forest Classifier

In order to automatically annotate videos, the process of annotation can be simplified to a binary classification problem between two possible classifications: yes, the behavior occurs and no, the behavior does not occur. The machine learning classifier that was chosen to perform the classification was the Random Forest Classifier. This specific model was chosen due to the relative success of prior automatic annotation systems using this classifier, as well as the ease at which the inner workings of the classifier can be visualized.

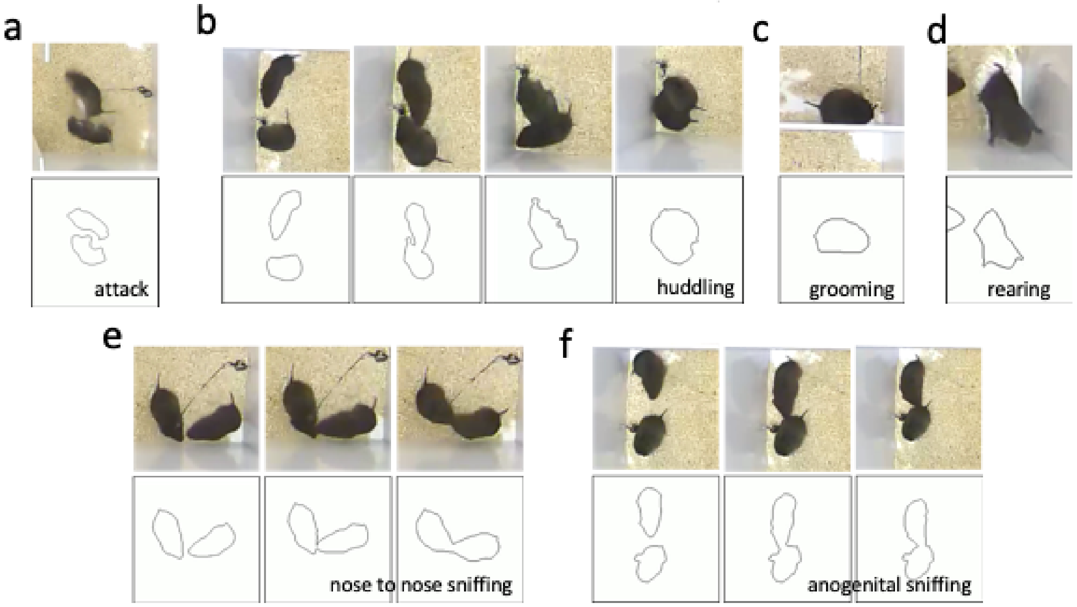

Figure 3: a) attack behaviors b) series of frames entering into a huddling behavior c) grooming behavior d)rearing behavior e) nose-to-nose sniffing/interaction f)angiogenital sniffing/interaction

The labels and features as described above were normalized in order for the classifier to have optimal performance. Missing values were filled using the mean of each feature. For features that were calculated over several frames, the initial frames had no values. Therefore, these values were also filled using the mean in order to decrease the range and increase the chances that the feature would be interpreted correctly by the classifier. Other values, such as eccentricity, were centered over a singular value to normalize data over all frames.

Multiple hyper-parameter combinations were tested using a grid search to determine which set of hyper-parameters would be the best suited for this classifier and the associated data. The final model had 200 trees, with the maximum depth of each tree set to 10. In addition, the number of data points in each leaf node was tuned to 2. The rest of the hyper-parameters were left as their default values. While there were many features calculated, some of the features would not be applicable for huddling behaviors because the contours merge together. For example, values such as angle of orientation cannot be calculated during huddles since major axis endpoints no longer correlate with head or rear regions. Therefore, feature selection narrowed down a smaller number of features that were hypothesized to have a higher feature importance due to trends seen visually in manually annotated videos. Some of these features include velocity, area and distance from corners. The effectiveness of the classifier was quantified using multiple different measures, which are summarized in Table 1. Precision focuses on the amount of values that were correctly classified as positive, and does not take into account false negatives. In contrast, recall does not take into account false positives. Therefore, the F1 score, which is the harmonic mean of both precision and recall, was chosen to quantify the effectiveness of each model.

Measure	Equation
Precision	$\frac{TP}{TP+FP}$
Accuracy	$\frac{TP+TN}{TP+FP+FN+TN}$
Recall	$\frac{TP}{TP+FN}$
F1	$\frac{2*(Recall*Precision)}{Recall+Precision}$

Table 1: A summary of the different measures used to determine the quality of a model. TP are true positives, FP are false positives, TN are true negatives and FN are false negatives.

3 Results

Datasets

There were 12 different videos used to develop the classifier. For each run, the dataset was split into training, testing and validation datasets using a 60 : 20 : 20 ratio. Out of the frames used in the classifier, 67% of frames contained either a ”huddle left” or ”huddle right” behavior for a majority to minority ratio of 67 : 32.

Behavioral Annotation Analysis

The manual annotation data was analyzed to examine the lengths of each behavior, as well as the percentage of time that the voles spent performing each behavior. The data is summarized in Figure 4. The behaviors annotated in these files include left, right, interact left, interact right, huddle left, huddle right and attack behaviors. However, attack behaviors are annotated as point behaviors, and therefore have no duration to be recorded. The length of huddles had a high standard deviation, as huddle lengths can vary greatly.

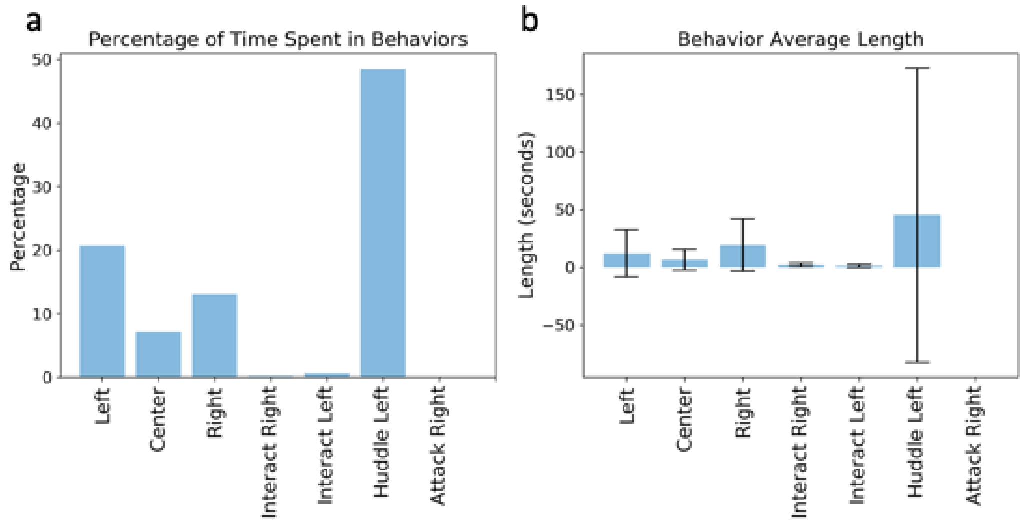

Figure 4: Behavior Average Lengths and Percentage of Time Spent in each behavior. Error bars represent the standard deviation of the dataset.

Model Performance

After splitting up the dataset into training, testing and validation, the model was trained on the training dataset. Using the validation dataset, the decision threshold was tuned according to the Precision-Recall Curve to find the point where both precision and recall were optimized to the highest value possible. The optimal threshold of the validation dataset was tuned to .61. The final model with the tuned hyper-parameters and threshold was tested on the final testing dataset.

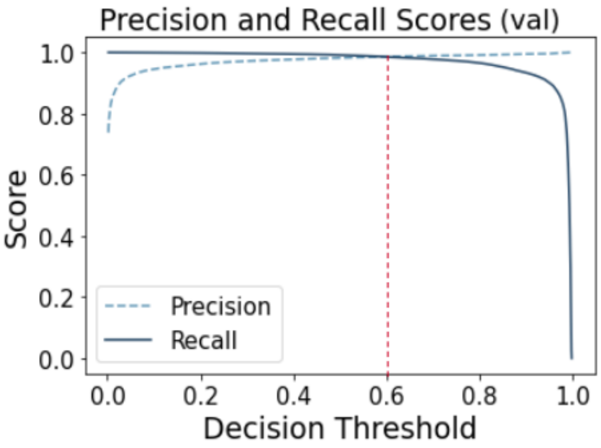

Figure 5: Precision and Recall Curve for Validation Dataset

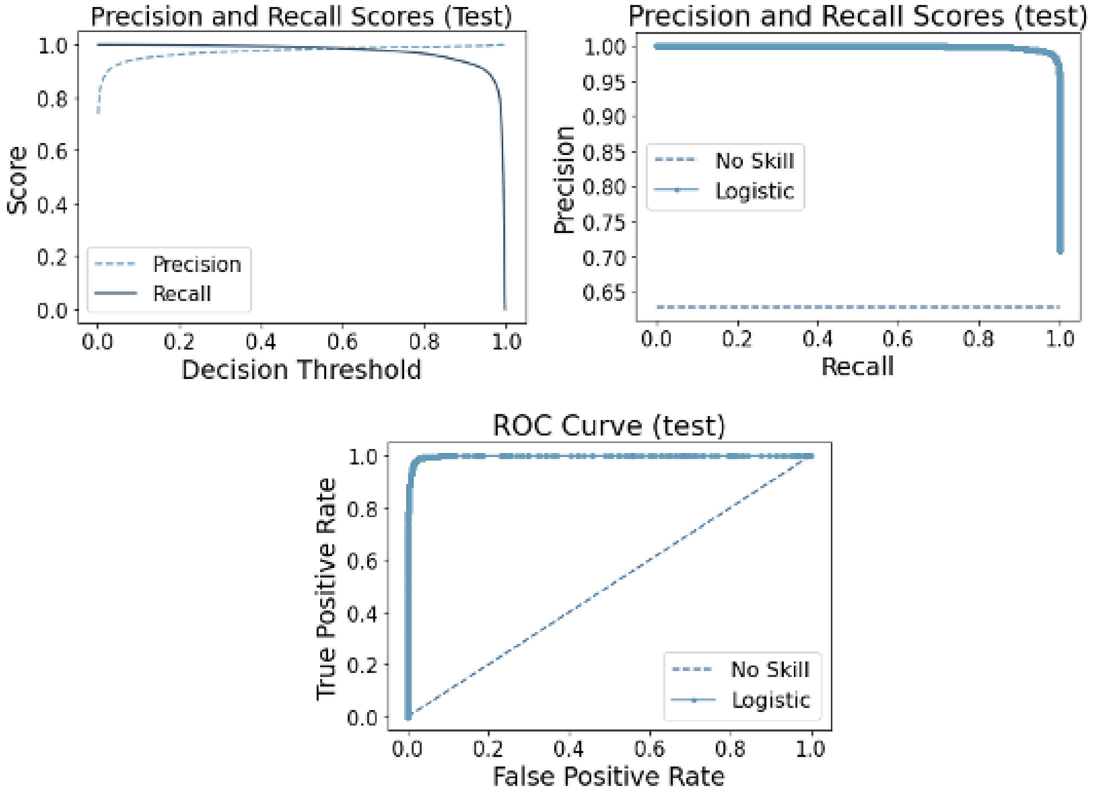

Figure 6: Precision and Recall and ROC Curves for Testing Dataset

Cross Validation

The model was evaluated using 5-fold cross validation, where different sets of the data was used for training and testing. This ensured that the model was tested in a variety of ways. The cross validation was run with the previously determined decision threshold of .61. The classification metrics were averaged, and are displayed in Table 3. The time taken for each run to finish training was also recorded, and averaged around 10 minutes.

	Precision	Recall	F1-Score
Absence of Behavior	97%	98%	97%
Presence of Behavior	99%	98%	98%
Accuracy	-	-	98%

Table 2: Performance Metrics for 5-fold Cross Validation

Feature Importance

Reducing the number of features can help increase the performance of a model, by reducing noise and allowing for more meaningful correlations to be made. Gini importance is a measure of how many times a specific feature is used to split a node, and is also affected by the number of samples that the node splits. [12]. The final model contained 8 main features, and their features were calculated using gini importance. The importances are summarized in Figure 7. The large standard deviations, as represented by the error bars, signify that these specific features carry different importances in different trees.

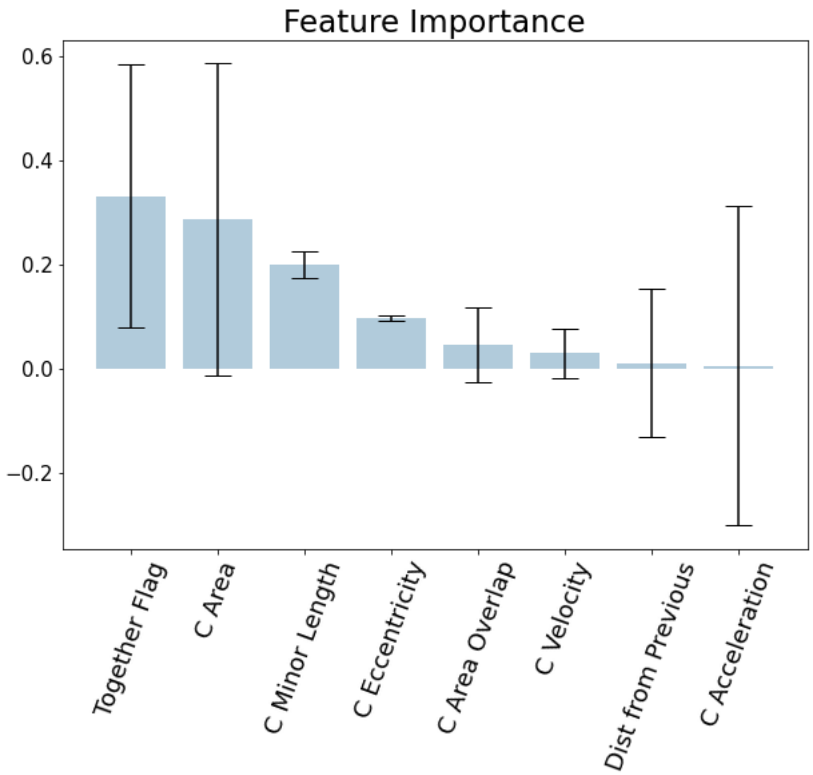

Figure 7: Gini Feature Importances

Scrambled Labels

5-fold cross validation was performed after scrambling the manual annotation labels and testing the model on a correct, unscrambled testing dataset. Scrambling the manual annotation brought down the classification metrics, as shown in Table 4.

	Precision	Recall	F1-Score
Absence of Behavior	44%	100%	83%
Presence of Behavior	0%	0%	0%
Accuracy	-	-	44%

Table 3: Performance Metrics for 5-fold Cross Validation with Scrambled Labels

False Positives and Negatives

The locations of the false positive and false negatives were analyzed to determine underlying causes and potential areas for model improvement. For example, some false positives were due to the classification of interactions as a huddling behavior, while others may even occur when there is no merged contour but the area of the contour was enlarged. These findings are summarized in the figure below.

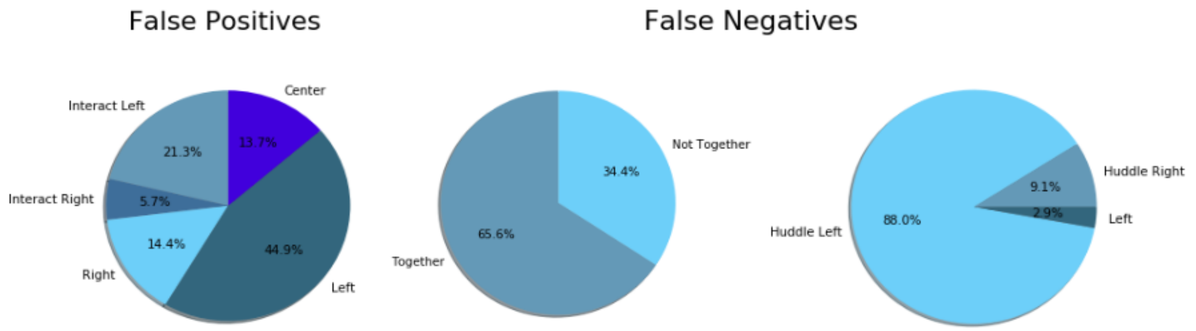

Figure 8: False positives were analyzed to determine the correct behavior. False negatives were analyzed to determine for which huddle type they occurred, as well as if a merged contour was present.

4 Discussion

Developing an automatic annotation system for the annotation of social behavior in voles is integral for promoting studies of social behaviors and their underlying causes. A classifier has been constructed for a social behavior known as huddling that is exclusively found in voles. The average F1 scores for the presence and absence of the behavior were 97% and 98% respectively. In addition, a variety of features from contour data that can be applied to detect other behaviors was calculated.

The most important features that were used by the model were the together flag, center vole area, center vole minor axis length and center vole eccentricity. The together flag was designed to distinguish between frames where there was a merged contour, as huddling behaviors are only possible in such cases. The eccentricity and minor-axis length were both hypothesized to have a high feature importance because of the nature in which the two voles huddle together. When the voles interact in a sniffing behavior, the contour is elongated as the two separate vole contours will merge face to face or face to rear as seen in Figure 4. In contrast, the huddling voles interact in a more rounded manner. Therefore, features such as eccentricity and minor axis length are important features in determining the behavior occurring. Vole area is also of importance since merged contours will have a higher area than unmerged contours. Since the left and right voles are tethered, the only possible huddle events can occur between the center and right or center and left vole. Therefore, the features used for the classifier was limited to features calculated on the center vole.

The overall class balance was a majority to minority ratio of 67 : 32. While the classes are not balanced completely, the balance is not skewed severely and therefore did not have an impact on classification metrics. Class balancing methods such as Synthetic Minority Over-sampling Technique (SMOTE) were applied to see if balanced classes can improve the efficiency of the classifier. [13] However, there was no significant difference in classification metrics which further shows that the class imbalances are not significant enough to affect performance. When the manual annotation labels were scrambled, the classification metrics dropped significantly. This decrease in scores shows that the classifier is depending on the training labels, and serves as a fail-safe to ensure that the model is working properly. However, there were high F-1 and recall scores for the absence of the behavior, which were due to the model classifying almost all frames as negatives.

The false positive and false negative analysis presented multiple insights into model performance. As seen in Figure 8, 34.4% of the false negative occurred when the vole contours were not together. However, a huddle behavior would not be expected without touching voles is not possible. This result shows that there are some discrepancies in human annotation, which is not 100% accurate. Human annotation can sometimes miss a few seconds at the beginning or end of a huddle, which the machine learning model can accurately predict. This opens up the possibility that the model may be more accurate than the current metrics suggest. A large percentage of the false positivse occurred when the recorded behavior was a left huddle. This result may be a consequence of the fact that left huddles are more common than the other social behaviors.

5 Future Work

The current performance metrics for the model can be optimized in multiple different ways. Currently, all features being used are contour-dependent features, which means that they have been calculated using the contour points. However, more accurate features can also be calculated using point data. Programs such as DeepLabCut, a deep learning approach to pose estimation, can be used to find nose, rear and other body points on the vole contour. [14] An initial challenge with using DeepLabCut was that it is not as effective for multiple animals, which is the primary reason why point-dependent features were not included in the current classifier. However, a new multiple animal version of DeepLabCut has recently been released, and work is currently being done to optimize the DeepLabCut software for vole assays. Once the point data has been calculated, there are many more features which can be created and may also help develop a more accurate classifier. Point data will also support future goals of pose estimation.

In addition, more elaborate feature engineering may also be necessary to extract the most important data from the current features in use. Currently, the main behavior of interest is huddling. However, there are many other behaviors that are also important in behavior studies. Therefore, future directions for this study include building classifiers for these other behaviors such as sniffing or attacks as these behaviors are also important social behaviors. More specific features for each behavior will be necessary in order to create individual classifiers. Some potential features include tortuoisity, which looks at how straight the path of a vole is, and the difference between the angle of orientation and angle of movement, as this can signify when the vole is not moving forward and could be moving sideways or backwards.

6 Conclusion

I have created an in-depth set of features for use in the automatic annotation of vole behavior from a set of contour data sourced from a set of vole studies. 21 main features were created for each contour in each frame, for a total of 138 features per frame. A classifier for a type of social behavior seen in voles, known as huddling, was developed using a random forest model. The average F1 scores for the presence and absence of the behavior were 97% and 98%, with an accuracy of 93%. Given the duration of model training when compared to the duration of manual annotation, an accuracy of 98% is relatively high. However, additional improvements can be made to improve this model further. These classifiers are integral for increasing the accessibility and accuracy of studies using voles and analyzing their social behavior. Future work will focus on improving current models and developing more models for other vole behaviors.

7 Acknowledgments

Thank you to Dr. Shuyu Wang, Ms. Kara Quine, River Grace, RSI, CEE, and MIT.

References

[1] L. J. Young, M. M. Lim, B. Gingrich, and T. R. Insel. Cellular mechanisms of social attachment, 01 2001.

[2] What is a mouse model? *The Jackson Laboratory.*

[3] A. K. Beery, J. D. Christensen, N. S. Lee, and K. L. Blandino. Specificity in sociality: Mice and prairie voles exhibit different patterns of peer affiliation.

[4] C. Sue Carter, A. Courtney Devries, and L. L. Getz. Physiological substrates of mammalian monogamy: The prairie vole model, 01 1995.

[5] J. M. Sadino and Z. R. Donaldson. Prairie voles as a model for understanding the genetic and epigenetic regulation of attachment behaviors, 01 2018.

[6] A neurobiological basis of social attachment, 01 1997.

[7] S. R. Datta, D. J. Anderson, K. Branson, P. Perona, and A. Leifer. Computational neuroethology: A call to action, 01 2019.

[8] W. Hong, A. Kennedy, X. P. Burgos-Artizzu, M. Zelikowsky, S. G. Navonne, P. Perona, and D. J. Anderson. Automated measurement of mouse social behaviors using depth sensing, video tracking, and machine learning, 01 2015.

[9] S. R. Nilsson, N. L. Goodwin, J. J. Choong, S. Hwang, H. R. Wright, Z. C. Norville, X. Tong, D. Lin, B. S. Bentzley, and N. Eshel. Simple behavioral analysis (simba) – an open source toolkit for computer classification of complex social behaviors in experimental animals.

[10] S. Russo and G. Hodes. Faculty opinions recommendation of jaaba: interactive machine learning for automatic annotation of animal behavior., 01 2013.

[11] R. Schafer. What is a savitzky-golay filter? [lecture notes], 01 2011.

[12] S. Nembrini, I. R. König, and M. N. Wright. The revival of the gini importance?, 01 2018.

[13] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: Synthetic minority over-sampling technique.

[14] A. Mathis, P. Mamidanna, K. M. Cury, T. Abe, V. N. Murthy, M. W. Mathis, and M. Bethge. Deeplabcut: markerless pose estimation of user-defined body parts with deep learning, 01 2018.