Transfer Learning for Rodent Behavior Recognition

M. Lorbach[1,2], R. Poppe[1], E.A. van Dam[2], L.P.J.J. Noldus[2] and R.C. Veltkamp[1]

[1]Utrecht University, Utrecht, The Netherlands. m.t.lorbach@uu.nl
[2]Noldus Information Technology, Wageningen, The Netherlands

Most automated behavior recognition systems are trained and tested on a single dataset, which limits their application to comparable datasets. While training a new system for a new dataset is possible, it involves laborious annotation efforts. We propose to reduce the annotation effort by reusing the knowledge obtained from previous datasets and adapting the recognition system to the novel data. To this end, we investigate the use of transfer learning in the context of rodent behavior recognition. Specifically, we look at two transfer learning methods and examine the implications of their respective assumptions on synthetic data. We further illustrate their performance in transferring a rat action classifier to a mouse action classifier. The performance results in the transfer task are promising. The classification accuracy improves substantially with only very few labeled examples from the novel dataset.

Introduction

The automated recognition of rodent behavior plays an important role in studies of neurological disorders such as Huntington's disease. With the introduction of automated recognition systems, the results of such studies have become more accurate and better reproducible across laboratories [1].

Automated behavior recognition systems typically track the animals in videos and classify their behavior into several action categories. This classification involves an action model that is typically learned from annotated training sequences. Often these training sequences originate from a single dataset of one specific study [2]–[6]. As a consequence, a model that is applied to a dataset that is different from the training dataset may cause unwanted bias in the results [7].

In practice, every dataset is different. The data are affected by a variety of factors that occur in animal studies. For instance, laboratories use different video acquisition setups and cages. The animals behave differently depending on their species, age, gender and individual traits, but also on their treatment and potential disease progress. Each of these factors has an effect on the data distributions and therefore potentially on the accuracy of the recognition system.

To ensure high recognition accuracy on a novel dataset, we could train a new action model tailored to that dataset [8]. However, the training comes at the cost of manual annotation efforts. Furthermore, a new action model would not take previous models into account and would therefore neglect valuable knowledge. Instead of learning a new action model from scratch, we propose to adapt the existing model to the new dataset by adopting a transfer learning approach.

The key idea of transfer learning is to build upon previously obtained knowledge and combine it with a small amount of additional training data to obtain a model that performs optimally with respect to a novel dataset. The additional training data may contain annotated but also unannotated instances from the novel dataset depending on the method. In order to be able to balance previous and new knowledge, transfer learning methods make varying assumptions about which parts of the model do and do not change across datasets.

In this paper, we take the first steps towards a behavior recognition system that combines previously learned knowledge with the ability to adapt to new datasets. We investigate the strengths and limitations of two transfer learning methods with fundamentally different assumptions in the context of rodent behavior recognition. We examine the implications of their respective assumptions on synthetic data and show their performance in a task transferring an action model trained on rats to a new action model for mice.

Related Work on Rodent Behavior Recognition

The automated recognition of rodent behaviors typically involves classifying video frames into behavior categories based on features extracted from the video. The features may contain the trajectories of the moving animals as well as shape, pose and distance information. The system's performance is eventually evaluated on a test set by comparing the automated classification to the annotations of one or multiple humans.

A large number of recognition systems for animal behavior are trained and evaluated on subsets of one dataset [2]–[6]. Using different strategies, the dataset is split into training and evaluation sets. To increase the statistical power of the evaluation, multiple splits can be generated randomly and the accuracy values are then averaged [9]. Despite their statistical power, the scope of reported performance is limited to one specific dataset and we are uncertain about the validity on other datasets.

The performance can be considerably lower on other datasets. An automated behavior recognition method for rats, for instance, has been reported to show a decrease in average precision by 17% if the system is tested on animals it has not encountered during training compared to when it was trained and tested on the same animal [10]. Along this line of thought, another study has recently shown that individual animals have distinct modes in their velocity and distance values [11]. Such individual traits potentially affect the automatic classification of, for example, *walking* and *running*.

In addition, the experiment environment can influence the performance. For instance, illumination and viewpoint variations may affect the animal tracking whereas the dimensions of the cage scales distance and velocity features. Although some features such as distances can be normalized [12], factors such as individual traits of the animals do not correspond to single features and are therefore difficult to compensate for manually. A possible countermeasure is to include the expected variations of the environment and the animals in the dataset [9], [12]. Although such variations can improve the generalization properties of a classifier, they are limited to the included variations. Clearly, it is infeasible to create an annotated dataset that includes the infinite number of real-world variations and then test on it efficiently. Moreover, data with high variance pose classification challenges that can only be countered by complex classifiers and even more data.

In this work we investigate the task of adapting a classifier that has been trained on one dataset to a novel dataset. In the field of machine learning such a task is considered a transfer learning problem.

Transfer Learning

Transfer learning addresses classification and regression scenarios in which the training and the test data are sampled from different data distributions [13]. Such scenarios occur frequently in real-world applications. For instance, for medical diagnosis a disease model may be learned from a group of affected patients and an equally sized group of healthy volunteers. If such a model is later applied to predict the disease in a much larger and more general population, it is likely to cause biased results. In this particular case, the model suffers from a sample selection bias.

Sample Selection Bias

Sample selection bias can degrade the accuracy of a model considerably [14]. We distinguish between two main sources of how a sample may become biased. First, as in the example of the disease model, the training set contains relatively more positive examples than the true population. The model is biased towards the positive diagnosis. In this case, the sample selection is dependent on the label (positive or negative).

Second, the sample selection can also depend on the feature values. Consider the task of classifying fish species based on their length. If we take the training examples from a pool of mainly juvenile fish, we obtain a distorted image of the true distribution of length values. Again, classifications we make on basis of that model are biased.

Once we are aware of a sample selection bias in our training data, we can attempt to correct for it using a transfer learning approach. In transfer learning we consider data coming from different domains. While labeled training data is obtained from the source domain, the test data comes from the target domain. The transfer learning problem is then formulated as the task of finding a classifier that performs optimally in the target domain.

Several methods address sample selection bias. The majority assumes that we do not have any labeled instances from the target domain to guide the knowledge transfer across the domains. Although we rely on training the model on source data, we can influence the training using the *unlabeled* target data. For instance, we can concentrate our learning effort on source samples that are surrounded by many target samples in the feature space and neglect the deviating samples. Assuming that the labels between the selected samples are the same, we yield a suitable model of the target domain. We need to ensure, however, that all target samples are close to at least some source samples so that all target samples are eventually reflected in the model.

The concept of identifying the most important source samples is the key idea behind Kernel Mean Matching (KMM) [15], [16]. KMM distributes weights to the source samples according to their importance by minimizing the difference between the means of the target data and the weighted source data. Since the weighted source data match the target data better, a classifier trained on the weighted data performs better in the target domain.

A similar approach is taken by Transfer Component Analysis (TCA) [17]. Instead of weighting source samples, TCA finds a feature mapping that minimizes the difference between source and target distributions in the mapped feature space.

Regardless of the source of the bias, a sample selection bias influences the marginal probability distributions. As a consequence the distribution over samples X differ between domains, i.e., $P_{Source}(X) \neq P_{Target}(X)$. The key assumption that enables us to compensate for these differences is that the conditional probability distributions are not affected by the bias. That is, the probability of the occurrence of a label y given a sample x is the same irrespective of the domain: $P_{Source}(y|x) = P_{Target}(y|x)$.

General Dataset Shift

In the case that the assumption about the conditional distribution does not hold, we are facing a more general dataset shift problem. To solve it, we typically need to make other assumptions about the domain differences [18] or introduce information about the target domain, for example, by providing labeled target instances [19].

The availability of labeled target instances enables us to estimate and optimize the performance of a classifier in the target domain. The transfer AdaBoost (trAdaBoost) [20] method repeatedly trains a classifier from the source data and predicts the labels of the known target instances. In each iteration it removes source instances that did not contain valuable information for classifying the target samples. After a number of repetitions the classifier is left with only the most important source samples and hence performs optimally in the target domain.

As opposed to manipulating training instances and feature representations to change the data model, we are also able to directly manipulate the classifier and its parameters. The practical advantage is that the original source data does not need to be available for the transfer [21]. The disadvantage is that introducing new labeled examples involves new challenges such as finding the right balance between the established source model and the new target model [22].

An example of a method that manipulates an existing classifier is the adaptive Support Vector Machine (aSVM) [23]. The prerequisite for aSVM is a classifier that has been trained on the source data. aSVM then changes the decision function of that classifier such that the classification error regarding the target instances is minimized. At the same time, the modification of the decision function is kept as small as possible. The balance between classification error and degree of modification is controlled via a parameter that is determined manually.

For the purpose of investigating transfer learning in the context of rodent behavior recognition, we concentrate our analysis on two methods that take two fundamentally different approaches: KMM and aSVM. First, we examine their performance on synthetic data with a particular focus on violating their key assumptions. Second, we apply them to a simplified rat to mouse behavior transfer problem and discuss their performance.

Experiments on Synthetic Data

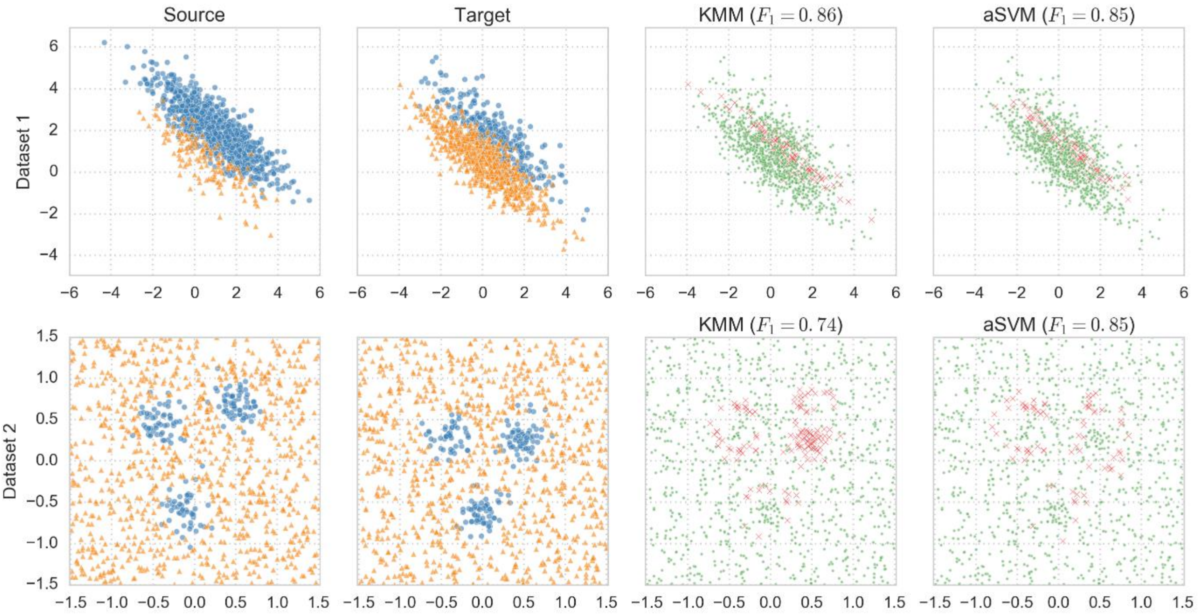

Figure 1: Left: Two synthetic datasets with two classes each are split into a source and a target set. Right: the correct (green dots) and incorrect (red crosses) predictions of KMM and aSVM (5% labeled target instances) on the target data.

We apply the two transfer learning methods, KMM and aSVM, to two synthetic datasets. The datasets are illustrated in Figure 1. Dataset 1 contains two classes each represented by a bivariate Normal distribution (class 1: $\mu = [1 \quad 2], \sigma = \begin{bmatrix} 1.7 & -1.2 \\ -1.7 & 1.7 \end{bmatrix}$, class 2: $\mu = [0.5 \quad 0.6], \sigma = \begin{bmatrix} 1.7 & -1.2 \\ -1.7 & 1.7 \end{bmatrix}$). The source dataset is sampled from the true distribution under a sampling bias as proposed in related work [14]. Samples of class 1 are selected with a probability of 0.85 ($N_1 = 850$) while samples of class 2 with a probability of 0.15 ($N_2 = 150$). The target dataset consists of $N_1 = 300$ and $N_2 = 700$ samples and the class means are shifted by $\Delta\mu = [-0.3 \quad -0.2]$. Dataset 1 therefore includes a sample selection bias and a small shift of the conditional probability distribution.

Dataset 2 has been used in related work to illustrate the aSVM method [23] and consists of a positive and a negative class. The positive class is represented by a mixture of three bivariate Normal distributions while the negative class is distributed uniformly outside the positive class. In the source domain, the positive class is determined by the parameters $\mu_1 = [-0.4 \quad 0.5], \mu_2 = [0.5 \quad 0.7], \mu_3 = [-0.1 \quad -0.6], \sigma = 0.02$; in the target domain by $\mu_1 = [-0.4 \quad 0.3], \mu_2 = [0.5 \quad 0.3], \mu_3 = [0 \quad -0.65], \sigma = 0.02$. Both source and target sets comprise 166 positive samples and 834 negative samples. Dataset 2 does not suffer from a sample selection bias but has a considerable shift in the conditional probability distributions.

Before training the datasets are scaled to zero-mean and unit-variance based on the source data only. KMM is applied to the source and the entire, unlabeled target data to obtain the sample weights for the source samples. A SVM is then trained on the weighted source samples with parameters for dataset 1: linear kernel, C=1; and for dataset 2: radial basis function (RBF) kernel, C=1, $\gamma = 5$.

For aSVM, we first train an SVM on the source data with the same parameters as before. Then, aSVM is applied to adapt the trained classifier given a varying amount of labeled target instances. We provide 1%, 2%, 3%, 4%

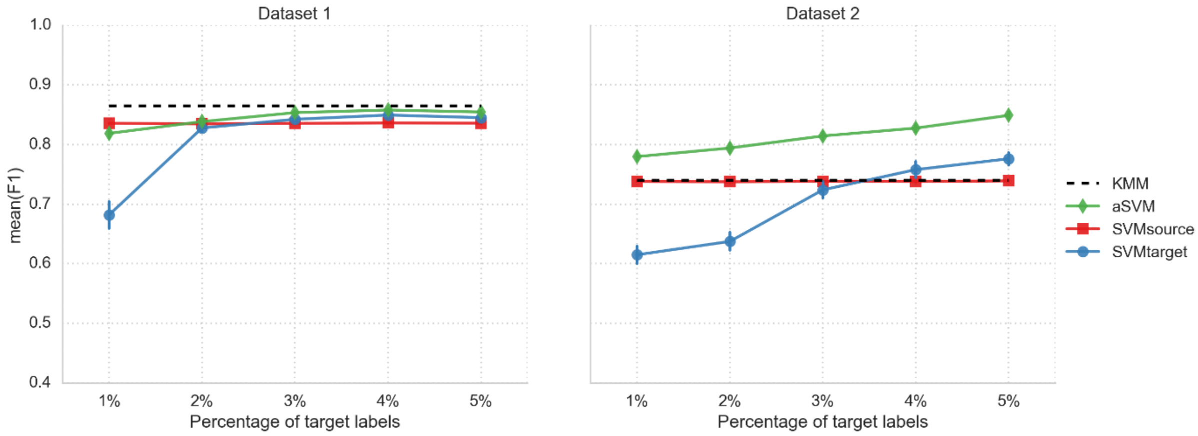

Figure 2: Accuracy on synthetic datasets with varying amount of labeled target instances. Error bars show the standard error of the mean. Number of target samples (100%): N=1000.

and 5% of the target set as labeled instances and subsequently test on the remaining samples. We perform 30 repetitions using randomized samples and average the results among the repetitions.

We evaluate the performance of the two methods in terms of the F1 score averaged across classes. For comparison, we include the performance of two baseline classifiers that do perform any knowledge transfer. First, an SVM is trained only on source data (SVMsource) and second, an SVM is trained only on the available target data (SVMtarget). All transfer learning methods should outperform both baseline classifiers.

Results

The averaged accuracies of the methods are shown in Figure 2. On dataset 1, KMM outperforms the other methods slightly. The accuracy of aSVM is comparable to SVMsource with only 1% of labeled target instances. With more data, aSVM converges to a similar decision boundary as KMM with comparable classification errors (Figure 1). If the SVM is trained on the labeled target data alone, we need at least 2% of the data to reach a performance comparable to the other methods.

On dataset 2, KMM does not improve the prediction accuracy over the SVM that is trained on the source samples. The SVMtarget outperforms SVMsource if it has access to more than approximately 3.5% of the target data. aSVM achieves the highest accuracy of the compared methods. The accuracy of aSVM increases as more target instances become available. The increase is particularly strong on this dataset. The source model can be improved even with very few target instances.

Discussion

KMM is able to compensate for the sample selection bias and the small shift in the features in dataset 1. Due to the weighing of source samples, the resulting classification model is a good fit for the target data. In contrast, on dataset 2, the assumption that the conditional distributions remain the same across domains is violated and consequently KMM misclassifies more samples. Eventually, the lack of labeled target instances prevents KMM from improving the original source model.

aSVM is able to use the available target labels to improve the source model. Moreover, aSVM outperforms SVMtarget which shows that the combination of knowledge from a previously trained model and knowledge from new instances is superior to learning a new model from scratch.

The experiments on synthetic data show that the KMM is too sensitive to violating its key assumptions. aSVM appears more robust to changes in the conditional distributions and may therefore be better suited for a real-world application such as rodent behavior recognition.

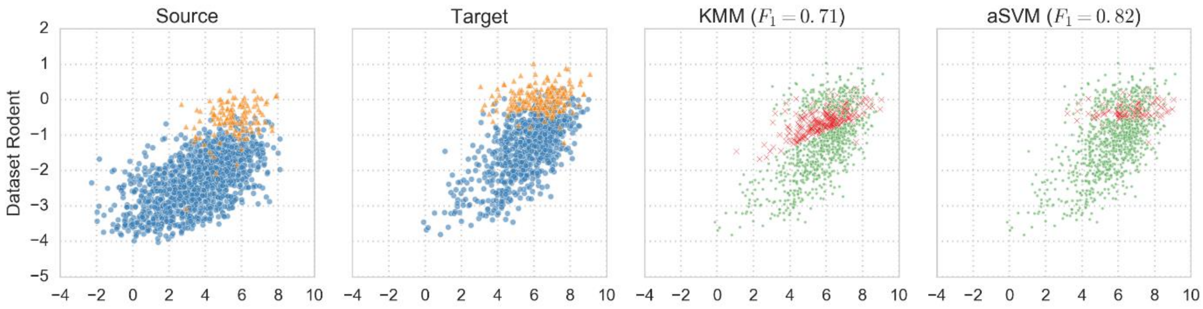

Figure 3: Left: Rodent behavior dataset with two classes *sniff* (blue circles) and *walk* (orange triangles). Right: the correct (green dots) and incorrect (red crosses) predictions of KMM and aSVM (1% labeled target instances) on the target data.

Experiments on Rodent Behavior Data

We now apply the two transfer learning methods to a simplified dataset of rodent behaviors. The learning task consists of transferring an action model for rats (source domain) to mice (target domain). Rats and mice perform similar actions but differ for example in their size and velocity. These differences affect the marginal as well as the conditional probability distributions.

The simplified dataset comprises a subset of a larger rodent behavior dataset. Every sample in the dataset corresponds to a video frame from which a number of features are extracted. For the purpose of this article we select two actions *sniff* and *walk*, and two features *Velocity* and *Optical Flow Energy*. The chosen subset reflects a scenario in which the shift in the conditional distribution causes misclassifications in practice. The source set contains $N_{sniff} = 1261$ (91%) and $N_{walk} = 127$ (9%) samples of sniffing and walking actions of one rat, respectively. The target set contains $N_{sniff} = 820$ (81%) and $N_{walk} = 194$ (19%) samples of one mouse.

The rodent dataset poses several classification challenges. The occurrence of the classes is highly unbalanced with a small selection bias towards *sniff* in the source domain. Furthermore, the conditional distributions differ between the domains as visualized in Figure 3.

We apply both KMM and aSVM analogous to the synthetic datasets. Source and target SVMs are parameterized with a linear kernel (C=1). The training is performed with a varying amount of labeled target instances and is repeated 30 times with randomized samples. The performance is evaluated in terms of the F1 score averaged over classes.

Results

The accuracy of aSVM increases with the number of available labeled target instances (Figure 4). Similar to the synthetic dataset 2, only five labeled samples are sufficient to reach a substantial increase in performance over both SVMsource and SVMtarget. With approximately 3.5% of the target instances being labeled, SVMtarget has sufficient information to perform almost as well as aSVM.

KMM improves the accuracy of the original source model by approximately 0.14 but is outperformed by aSVM with only 0.5% labeled target instances.

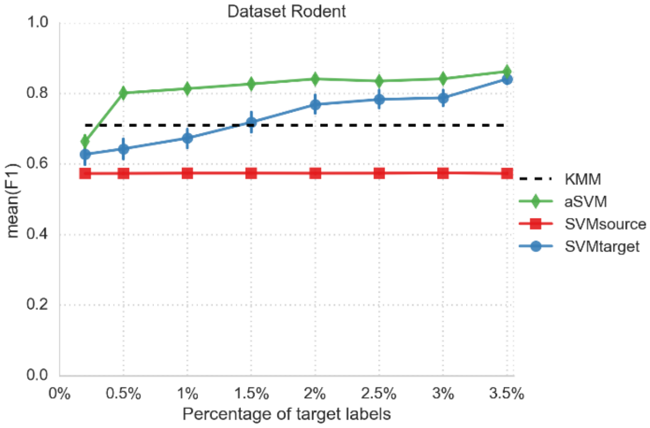

Figure 4: Performance on real rodent behavior data with varying amount of labeled target instances. Error bars show the standard error of the mean. Number of target samples (100%): N=1014.

In Figure 3 we count more misclassifications by KMM compared to aSVM. aSVM manages to find a decision boundary that matches the classes in the target domain well. A few classification errors remain along the decision boundary.

Discussion

The main challenge of the rodent dataset is similar to the challenge in synthetic dataset 2, namely the shifted conditional distribution. Consequently, aSVM is, as in the synthetic case, able to deal with that challenge better than KMM. aSVM therefore presents a viable option for transferring classifiers to another domain even if only very few labeled target instances are available.

In all analyzed datasets, the transfer of knowledge from the source to the target domain is achieved best by aSVM. Although it cannot quite match the accuracy of KMM on the synthetic dataset 1, aSVM is more efficient to compute. While KMM evaluates the pairwise distances between all source and target samples, the calculations for aSVM are limited to minimizing the expected loss over the small amount of labeled target samples. The efficiency of aSVM will be an advantage if knowledge is to be transferred among larger datasets.

In the performed experiments, we posed relatively simple classification problems. The classifiers' performances saturate with very few examples so that the differences between the models vanish quickly. Moreover, we have only looked at binary classification tasks in a two-dimensional feature space. In rodent behavior recognition we need to distinguish between 8 or more classes in much higher dimensions. Further investigations are needed in order to determine the limits of aSVM.

A practical aspect that requires our attention is a suitable strategy for sampling labeled target instances. In this work, we sampled uniformly from the available target data. For rodent behavior recognition, annotating single, random video frames may not be the best strategy as it is important to capture the variance within the performance of an action as well as across such performances. Therefore, annotating multiple, continuous segments of different behaviors may be a better approach.

A related question is whether we can make the learning more efficient by avoiding random sampling and rather focusing on the most informative segments first. If this selection process could be automated then the user could be queried to provide annotations for the informative segments. The adaptation of the recognition system to a new dataset could then be carried out interactively with the user.

Conclusion

We have investigated transfer learning in the context of rodent behavior recognition. With transfer learning we aim at exploiting previously obtained knowledge from other datasets in order to improve the recognition in a novel dataset while reducing the annotation effort. In this article, we have evaluated two transfer learning methods and examined the implications of their respective assumptions on synthetic data. We have further illustrated their performance in transferring a rat action classifier to a mouse action classifier.

The results of aSVM on the rodent behavior transfer task are promising. The fact that very few labeled examples from the novel dataset are sufficient to substantially improve the classification accuracy, encourage further investigations using aSVM. In the future, we will evaluate its performance on more complex data with more classes and in higher dimensional feature spaces.

In its current implementation, aSVM can only handle binary classification problems. For its application to a wider range of rodent behavior recognition problems, the method has to be extended to multi-class problems. We will concentrate our efforts on extending the approach to more realistic classification tasks.

With our investigations, we have made the first step in the development of a rodent behavior recognition system that is adaptive to novel datasets under reduced annotation efforts. Further steps will be taken to enhance the applicability to more complex problems and to include the interaction with the user for higher efficiency.

Acknowledgement

This work was funded by the European Commission under the 7th Framework Programme (FP7) as part of the Marie Curie Initial Training Network ``PhenoRat" (GA no. 317259).

References

[1] F. A. Desland, A. Afzal, Z. Warraich, and J. Mocco, “Manual versus Automated Rodent Behavioral Assessment: Comparing Efficacy and Ease of Bederson and Garcia Neurological Deficit Scores to an Open Field Video-Tracking System,” *J. Cent. Nerv. Syst. Dis.*, vol. 6, pp. 7–14, 2014.

[2] H. Dankert, L. Wang, E. D. Hoopfer, D. J. Anderson, and P. Perona, “Automated monitoring and analysis of social behavior in Drosophila,” *Nat. Methods*, vol. 6, no. 4, pp. 297–303, 2009.

[3] E. Eyjolfsdottir, S. Branson, X. P. Burgos-Artizzu, E. D. Hoopfer, J. Schor, D. J. Anderson, and P. Perona, “Detecting Social Actions of Fruit Flies,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014, vol. 8690, pp. 772–787.

[4] 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,” *Proc. Natl. Acad. Sci.*, vol. 112, no. 38, pp. E5351–E5360, 2015.

[5] X. P. Burgos-Artizzu, P. Dollár, D. Lin, D. J. Anderson, and P. Perona, “Social behavior recognition in continuous video,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 1322–1329.

[6] L. Giancardo, D. Sona, H. Huang, S. Sannino, F. Managò, D. Scheggia, F. Papaleo, and V. Murino, “Automatic Visual Tracking and Social Behaviour Analysis with Multiple Mice,” *PLoS ONE*, vol. 8, no. 9, p. e74557, 2013.

[7] J. M. Girard and J. F. Cohn, “A Primer on Observational Measurement,” *Assessment*, 2016.

[8] M. Kabra, A. A. Robie, M. Rivera-Alba, S. Branson, and K. Branson, “JAABA: interactive machine learning for automatic annotation of animal behavior,” *Nat. Methods*, vol. 10, no. 1, pp. 64–67, 2012.

[9] H. Jhuang, E. Garrote, X. Yu, V. Khilnani, T. Poggio, A. D. Steele, and T. Serre, “Automated home-cage behavioural phenotyping of mice,” *Nat. Commun.*, vol. 1, no. 6, pp. 1–9, 2010.

[10] J. B. I. Rousseau, P. B. A. Van Lochem, W. H. Gispen, and B. M. Spruijt, “Classification of rat behavior with an image-processing method and a neural network,” *Behav. Res. Methods Instrum. Comput.*, vol. 32, no. 1, pp. 63–71, 2000.

[11] S. M. Peters, I. J. Pinter, H. H. J. Pothuizen, R. C. de Heer, J. E. van der Harst, and B. M. Spruijt, “Novel approach to automatically classify rat social behavior using a video tracking system,” *J. Neurosci. Methods*, 2016.

[12] E. A. van Dam, J. E. van der Harst, C. J. F. ter Braak, R. A. J. Tegelenbosch, B. M. Spruijt, and L. P. J. J. Noldus, “An automated system for the recognition of various specific rat behaviours,” *J. Neurosci. Methods*, vol. 218, no. 2, pp. 214–224, 2013.

[13] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” *Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2010.

[14] B. Zadrozny, “Learning and Evaluating Classifiers under Sample Selection Bias,” in *International Conference on Machine Learning (ICML)*, 2004, pp. 903–910.

[15] J. Huang, A. Smola, A. Gretton, K. Borgwardt, and B. Schölkopf, “Correcting Sample Selection Bias by Unlabeled Data,” in *Advances in Neural Information Processing Systems*, 2007, pp. 601–608.

[16] M. Sugiyama, T. Suzuki, S. Nakajima, H. Kashima, P. von Bünau, and M. Kawanabe, “Direct importance estimation for covariate shift adaptation,” *Ann. Inst. Stat. Math.*, vol. 60, no. 4, pp. 699–746, 2008.

[17] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, “Domain adaptation via transfer component analysis,” *Neural Netw.*, vol. 22, no. 2, pp. 199–210, 2011.

[18] K. Zhang, B. Schölkopf, K. Muandet, and Z. Wang, “Domain Adaptation under Target and Conditional Shift,” in *International Conference on Machine Learning (ICML)*, 2013, pp. 819–827.

[19] X. Wang and J. Schneider, “Flexible transfer learning under support and model shift,” in *Advances in Neural Information Processing Systems*, 2014, pp. 1898–1906.

[20] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, “Boosting for Transfer Learning,” in *International Conference on Machine Learning (ICML)*, 2007, pp. 193–200.

[21] I. Kuzborskij and F. Orabona, “Stability and Hypothesis Transfer Learning,” in *International Conference on Machine Learning (ICML)*, 2013, pp. 942–950.

[22] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, “A Theory of Learning from Different Domains,” *Mach Learn*, vol. 79, no. 1–2, pp. 151–175, 2010.

[23] J. Yang, R. Yan, and A. G. Hauptmann, “Cross-domain Video Concept Detection Using Adaptive SVMs,” in *International Conference on Multimedia (MM)*, 2007, pp. 188–197.