Eurographics Workshop on Visual Computing for Biology and Medicine (2018)
I. Hotz, B. Kozlikova, and P.-P. Vazquez (Editors)

Short Paper

Introducing CNN-Based Mouse Grim Scale Analysis for Fully Automated Image-Based Assessment of Distress in Laboratory Mice

M. Kopaczka[1], L. Ernst[2], J. Schock[1], A. Schneuing[1], A. Guth[1], R. Tolba[2] and D. Merhof[1]

[1]Institute of Imaging and Computer Vision, RWTH Aachen University, Germany
[2] Institute of Laboratory Animal Research, RWTH Aachen University, Germany

Abstract

International standards require close monitoring of distress of animals undergoing laboratory experiments in order to minimize the stress level and allow choosing minimally stressful procedures for the experiments. Currently, one of the the best established severity assessment procedures is the mouse grimace scale (MGS), a protocol in which images of the animals are taken and scored by assessing five key visual features that have been shown to be highly correlated with distress and pain. While proven to be highly reliable, MGS assessment is currently a time-consuming task requiring manual video processing for key frame extraction and subsequent expert grading. Additionally, due to the the high per-picture expert time required, MGS scoring is performed on a small number of selected frames from a video. To address these shortcomings, we introduce a method for fully automated real-time MGS scoring of orbital eye tightening, one of the five sub-scores. We define and evaluate the method which is centered around a set of convolutional neural networks (CNNs) and allows live continuous MGS assessment of a mouse in real time. We additionally describe a multithreaded client-server architecture with a graphical user interface that allows convenient use of the developed method for simultaneous real-time MGS scoring of several animals.

1. Introduction and Previous Work

The 3R principle (replace, reduce, refine) first stated in [RB59] is a cornerstone of current laboratory animal experiment design and has found its way into international guidelines and standards for laboratory experiments with animals. It requires replacing live animal experiments where possible, reducing the number of animals required for obtaining the results and refining the experiment protocol for minimization of distress experienced by the animals undergoing the experiments. As a direct consequence, following the 3R principle requires methods for quantitative and objective scoring of animal pain and stress. Therefore, several methods for distress measurement have been developed. A large number of these methods focuses on manual or fully-/semi-automatic assessment of changes in animal behavior that correlate with distress; recent overviews can be found in [TTK14] and [DDV17]. Next to methods based on facial expression analysis these protocols also include analysis of other changes in animal behavior such as the approaches presented in [Jir14].

In recent years, distress recognition research has focused on observing facial areas such as eyes in order to develop reliable stress detection protocols. The work has been inspired by research carried out on humans, for which psychologists were able to show that a universal face of pain exists [Prk09]. Further research has shown that this facial expression is controlled by mechanisms that make it cross-cultural and even remain detectable in patients with dementia [KSH*07], indicating its fundamental nature. Inspired by this work, morphological changes in the facial appearance of different animals have been analyzed, resulting in a substantial number of findings. As a result, a number of grimace scales for different animals has been proposed, starting with the mouse grim scale (MGS) for laboratory mice [LBC*10] that has been proven to be highly reliable in practical use [ML15] and followed by similar research published for pain assessment in rats [WH14] [SSZ*11], rabbits [KTFL12] and ferrets [RSP*17], but also in larger animals such as horses [DCML*14], sheep [MRC*16] and pigs [DGBS*16].

As can be seen from these papers, scores based on facial expressions are gaining attention in the scientific community. They offer several advantages over other quantitative pain assessment methods: They do not require dedicated equipment such as the device required for gait analysis, and time requirements for the experiment itself are lower than for other behavioral experiments. Additionally, no extensive training is required to allow observers to learn how to perform grimace coding. As a downside, acquisition, selection and scoring of images is currently a time-consuming task. First approaches for automating this task have been proposed in [SSZ*11], where a face detection algorithm has been applied to detect rat faces in images. Automated facial expression analysis of sheep has been introduced in [LMR17] by applying a HOG-SVM based classification for automated facial feature analysis. A recent application of machine learning methods for automated pain classification in rodent images has been published in [TMJ*18]. In the aforemen-

DOI: 10.2312/vcbm.20181234

tioned paper, a convolutional neural network for binary classification is trained on a set of pain and non-pain images of white laboratory rats and it is shown that a general prediction of distress and non-distress states is possible when using neural networks for image classification.

In our work, we also focus on automated distress assessment in rodents using convolutional neural networks, however we extend the currently used approaches both regarding methodology and scope of applications. In more detail, the novelties of our contribution are:

- Use of current state-of-the art convolutional neural networks for semantic segmentation and region proposal, allowing precise subsequent classification of relevant image regions.
- Using actual MGS scores as reference, thereby enabling not only pain vs. no-pain classification, but also MGS score prediction. This allows an improved comparison to expert scores and also an improved consistency with currently established scores.
- Our method is trained and evaluated on a strain of black laboratory mice (C57BL/6), one of the globally most widely used mouse strains. Their dark fur color drastically limits the number of suitable methods that can be used for classification. For example, we implemented the methods described in [LMR17] based on HOG-SVM and [SSZ*11], where a Viola-Jones cascade classifier has been applied for face detection. Both methods have shown poor performance on images of C57BL/6 mice, with the HOG-SVM failing to predict pain state with an accuracy higher than chance level and the Viola-Jones detector being not able to perform face detection at all.
- Our method allows convenient automated real-time classification of images acquired live from a video stream. In this way, we obtain not only single image scores for distinct time points, but a continuous score in the time domain, allowing detailed classification of pain state of several simultaneously filmed mice in each video frame.

The paper is structured as follows: After this introduction and literature overview, we give a detailed overview of our method for image segmentation and classification as well as the client-server architecture that allows live scoring in video streams in Section 2. The methods are evaluated in Section 3, followed by a result discussion in Section 4 and a final conclusion in Section 5.

2. Materials and Methods

In this section, we describe the image acquisition setup, the neural networks used for segmentation and classification and the client-server based architecture for live scoring.

2.1. Imaging Setup Overview

For image acquisition, small transparent boxes following the descriptions in [LBC*10] were designed to allow recording the animals under controlled and replicable conditions. For increased efficiency, the boxes were placed in a rack holding up to four boxes. To ensure homogeneous ambient lighting and reduce reflections, the rack was placed in a light tent. A red background was chosen to allow increasing image contrast, as preliminary experiments have indicated that facial regions such as eyes can be best recognized in the red channel of the images. The setup was filmed with a HD camera at a resolution of 1080 x 1920 pixels at 30 frames per second. Figure 1 shows a sample frame from one of the videos.

Figure 1: *An image acquired with the imaging setup designed for our experiments*

2.2. Segmentation and Region Detection

Segmentation of the animals (foreground/background-extraction) was performed on subregion region-of-interest (ROI) images containing single boxes. These subregions can be defined either manually or using an automated method such as the box detection proposed in [KEH*18]. Using single box ROIs increases input consistency and allows using the method on experiment setups that do not use the multi-box design described above. For region segmentation, a U-Net [RFB15] fully convolutional network with shortcuts was used, which has shown excellent segmentation performance in terms of both run-time and segmentation accuracy in numerous challenging segmentation tasks.

Segmentation masks were designed with four classes - background, animal, ears and eyes. The distinct 'eye' class was chosen to allow using the U-Net as region proposal network for subsequent eye classification on smaller, equal-sized image patches. Ears were segmented as well to allow analyzing ear-related MGS subscores in the future. Figure 2 shows examples of segmentation results.

2.3. Automated MGS Scoring

For analyzing if automated MGS scoring is possible, we focused on the orbital tightening sub-score. The original MGS scale has discrete values of 0 (not present), 1 (moderate) and 2 (severe) for each

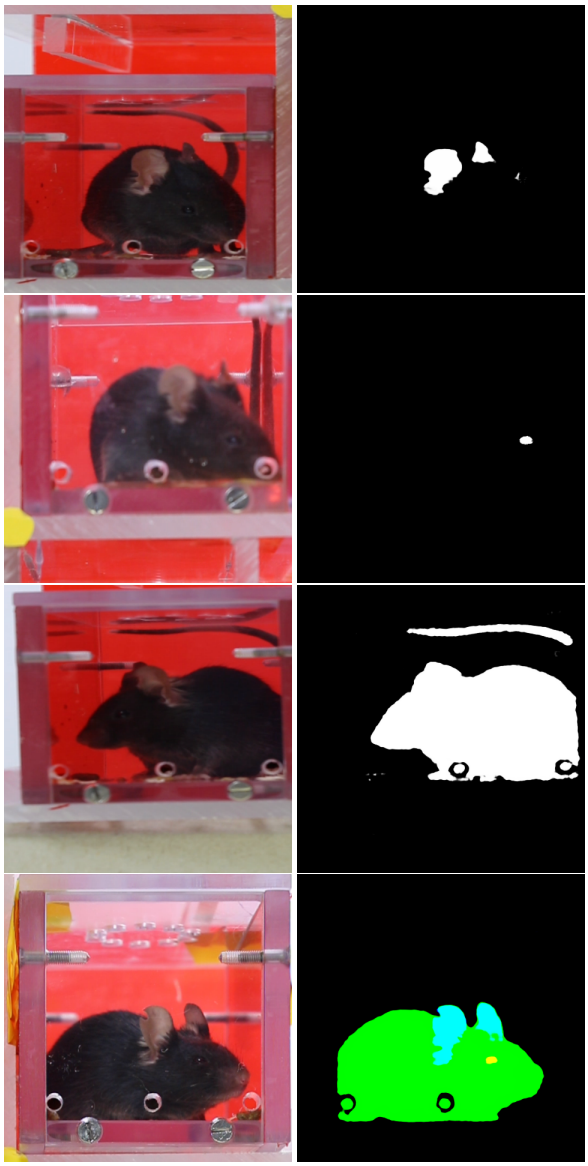

Figure 2: *Sample images and predictions acquired with the U-net. From top to bottom: ears, eyes, whole body and multilabel U-Net*

sub-score. For increased precision, we re-labeled the images with a more fine-grained scale between 0 and 9, with 0 corresponding to absence of the feature and 9 to maximal severity. We have decided to use the 10-point scale as initial experiments have shown that while introducing a larger number of classes may lead to increased label noise, the higher granularity of the input values reduces the impact of differently labeled images on the network training process. As example, if two virtually identical images are both on the edge between two values according to the rater and one of them is labeled with a 0 and the other one with a 1 on the three point scale during expert rating, then the label noise induced by this labeling mismatch is more confusing for the network than in the case of inconsistent labeling of these two images on the ten point scale, where one may have been labeled as 3 and the other as 4. We are currently gathering more expert data with both labeling strategies to analyze this hypothesis in more detail.

The convolutional neural network for predicting the orbital tightening score is a lightweight CNN containing two convolutional layers and two fully connected layers at its core mixed with auxiliary regularization layers for pooling, batch normalization and dropout (see Tab. 1 for the exact architecture). The network accepts image patches with a size of 50 x 50 pixels as input. The output prediction is formed by a single linear neuron allowing regression of the MGS score. While MGS scores themselves are discrete, we still decided to use regression instead of multi-class classification due to the continuous nature of our problem. The network design was focused on using a low number of parameters as analyzing orbital tightening is a task that should be addressable by a non-complex architecture which at the same time does not tend to overfit, a problem that more complex networks are prone to when using only a small number of training samples.

Layer Architecture				
Layer Type	Size	Stride	C/N	Act
Conv2D	(5, 5)	(1, 1)	6	ReLU
MaxPool2D	(2, 2)	(2, 2)	-	linear
Conv2D	(5, 5)	(1, 1)	16	ReLU
MaxPool2D	(2, 2)	(2, 2)	-	linear
AdaptiveAvgPool2d 5×5	-	-	-	linear
Flatten	-	-	-	-
Dense	-	-	120	linear
BatchNorm1D	-	-	-	ReLU
Dropout 0.25	-	-	-	-
Dense	-	-	84	linear
BatchNorm1D	-	-	-	ReLU
Dense	-	-	1	linear

Table 1: *Regression net architecture. C/N - Channels (for conv layers) or neurons (for fully connected/dense layers), Act - activation function*

2.4. Client-Server Architecture

We have integrated the algorithms for mouse segmentation and classification into a convenient graphical user interface. As current workstations in medical labs usually lack specialized GPUs that significantly speed up the execution time of neural networks, we decided to implement a client-server architecture that performs all computationally expensive tasks in a separate process that can be run either on the same computer or on a remote machine with more processing power. The communication link between both processes is established using ZeroMQ [Akg13], a cross-plattform middleware allowing inter-process communication between hosts implemented in several programming languages and potentially operating on different operating systems. ZeroMQ's support for both Windows and Linux allowed implementing the front-end on Windows - being a commonly used OS for computers in animal labs - and the classification back-end in Linux, which on the other side is commonly used for cluster machines with multiple GPUs. Using Linux also allowed using the powerful PyTorch library [PGC*17]

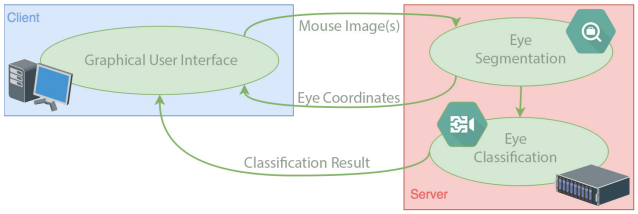

Figure 3: *The distributed architecture of our proposed system.*

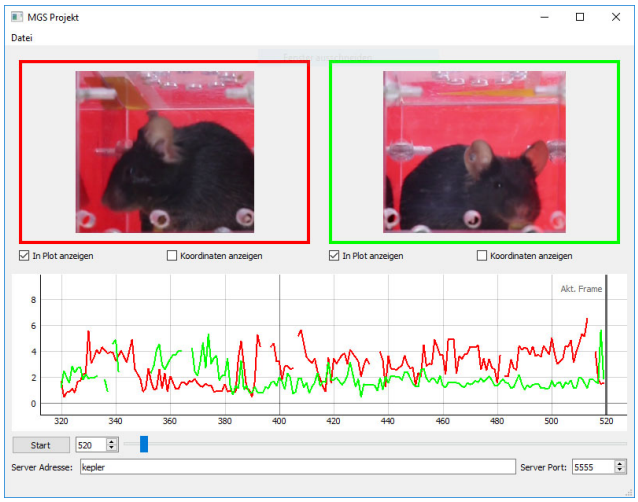

Figure 4: *Screenshot of the frontend's GUI*

for implementing the deep learning networks as PyTorch was not available for Windows at the time of implementation.

A schematic overview of our architecture is shown in Fig. 3. First, the user opens a video file and can select up to four ROIs contaninig mice for segmentation and classification. The ROIS are fed to the server in real-time at 30 frames per second. The CNNs perform segmentation and classification and return the center of the segmentation mask and the predicted MGS score to the client, where the data is used to display the analyzed ROIs seperately and optionally a marker located at the center of each detected eye. Additionally, the MGS scores are plotted continuously over time at the bottom of the user interface window. A screenshot of the GUI applied to two mice is shown in Fig. 4.

3. Experiments and Results

Here we present and evaluate our results for the previously described methods.

3.1. MGS Scoring

For actual MGS prediction based on eye patches, we trained our regression CNN with 314 manually labeled images. Evaluation was performed in a 30-fold cross validation. Due to the small number of labeled examples, we designed an augmentation strategy based on random subsampling: First, the training images are rotated by a random amount between +30 and -30 degrees. Subsequently, a 60 x 60 pixel sub-image centered at the eye center is extracted from the image. Finally, to account for the fact that the eye may possibly be not localized precisely by the segmentation and eye detection stage, we extract a 50 x 50 pixel patch from the sub-image, thereby simulating a shift by up to 5 pixels in any direction - a sufficiently high distance to cover realistic localization outcomes as indicated by the results of the segmentation described above. All these augmentation steps were performed during training on-the-fly using random parameters, i.e. the augmentation was random at each training run. Additionally, an analysis of the class distribution had shown that classes at the extreme ends of the score were strongly under-represented in the data set. This had been expected since the experiments performed on the mice were expected to produce 'moderately severe' results, therefore with only few mice showing no effects and at the same time only a few individuals displaying severe effects. To account for the uneven distribution, we added a sampling routine that adjusted the probability of a picture from a certain class to be fed into the network for training to be anti-proportional to the class probability of the image's label. Training was performed using random batches with a batch size of 32 for 35 epochs, using SGD optimization at a learning rate of 0.001, MSE loss and a momentum of 0.9. To additionally increase classifier performance and leverage for the small training set, we made use of pre-training for transfer learning. To this end, the net was pre-trained using the CIFAR-10 dataset [Kri09]. To allow multi-class classification instead of regression, we replaced the last layer with ten output neurons for the ten CIFAR-10 classes and trained the net using categorical cross-entropy. After finishing pre-training, the weights of the convolutional layers were transferred to the original regression network and training continued on the mouse eye data.

Figure 5 shows a Bland-Altmann plot [BA86] of the regression results compared with the manually given ground truth. This plotting method conveniently compares the mean and the difference of results returned by two methods (in our case the true and the predicted label) and therefore allows a graphical comparison of the method's properties. We predict a continuous value for discretely given ground truth values; this results in the linear clusters visible in the figure that allow a quick comparison of real and predicted values. The method's mean absolute error (MAE) is 0.871; the plot shows that the network has a slight tendency to underestimae very high and to overestimate very low labels. This may be partly due to the fact that the regression is bound to be within the margins of [0, 9]. The mean difference is -0.18, indicating a marginal underestimation of MGS values by our network.

3.2. Architecture Benchmark

We benchmarked our system on a client with the GUI front-end running on workstation hardware (Intel i5-4460, 4 cores at 3.2 GHz, 32 GB RAM) and connected it to a GPU server running using hardware for complex computations (Intel Xeon E5-2697, 18 cores at 2.3 GHz, 256 GB RAM, 8x nVidia GeForce GTX 1080 Ti) where segmentation and regression were performed. All video data was loaded by the client and the ROIs were forwarded to the server using ZeroMQ. The system performs live MGS scoring at

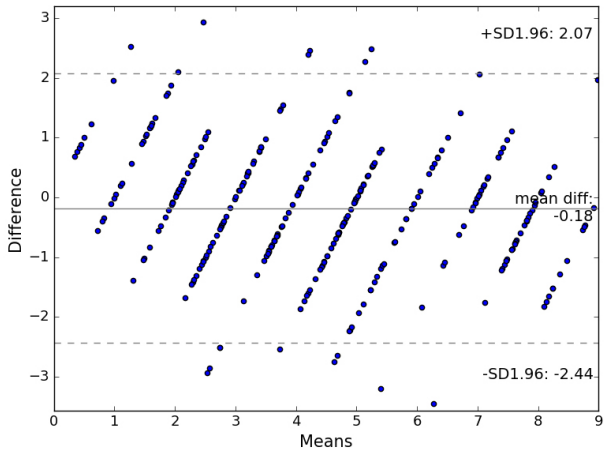

Figure 5: *Bland-Altmann plot of the classification results. The linear clusters are result of the discrete expert annotations that only used whole numbers as MGS values while the network outputs a continuous value.*

30 fps for up to three ROIs, with the frame rate dropping to 20 fps when scoring four animals. We assume that a multi-GPU approach that allows distributing the computations on several graphic cards would allow live scoring of four or more ROIs at full frame rate.

4. Discussion

The MAE of <1 being achieved by the classification network indicates that the method is a viable approach for mouse pain quantification using image data. We assume that the inter-rater variability between two human observers is within the same range; additional labeling experiments with multiple experts that will allow testing this assumption are currently on-going. The high frame rate of our implementation including all steps necessary for video preprocessing and region detection allows novel approaches towards pain quantification. Up to now, pain was measured by using selected frames from videos that may not capture the full range of expressions shown or introduce a bias depending on the frame selection strategy. With pain information being available continuously for every frame, novel experiment settings and evaluation strategies can be considered to take advantage of high-frequency real-time pain scoring.

5. Conclusion and Future Work

We presented a multi-stage method based on convolutional neural networks for live scoring of facial expressions in black laboratory mice. The approach is implemented using a server-client model and allows real-time MGS scoring on videos. It is extendable to videos of multiple animals and offers a graphical user interface for convenient use by non-technical staff. Evaluation of the segmentation stage shows that the implemented architecture allows precise mouse segmentation and eye detection for further processing. The classification using a fast CNN architecture shows a high consistency with manual annotations. We believe that after some refinement steps the software can be used for reproducible and quantitative automated pain estimation in laboratory mice under routine experimental settings. Next to usability and stability improvements, adding and evaluating algorithms for the automated assessment of the remaining MGS sub-scores can be considered. Moving from pain scores measures based on analysis of single images and averaging the results for a whole video to an individual score for every time point of the video will enable novel experiment setups, however methods for analyzing of the now continuous MGS data need to be developed as well.

References

[Akg13] AKGUL F.: *ZeroMQ*. Packt Publishing, 2013. 3

[BA86] BLAND J. M., ALTMAN D.: Statistical methods for assessing agreement between two methods of clinical measurement. *The lancet 327*, 8476 (1986), 307–310. 4

[DCML*14] DALLA COSTA E., MINERO M., LEBELT D., STUCKE D., CANALI E., LEACH M. C.: Development of the horse grimace scale (hgs) as a pain assessment tool in horses undergoing routine castration. *PLoS one 9*, 3 (2014), e92281. 1

[DDV17] DEUIS J. R., DVORAKOVA L. S., VETTER I.: Methods used to evaluate pain behaviors in rodents. *Frontiers in molecular neuroscience 10* (2017), 284. 1

[DGBS*16] DI GIMINIANI P., BRIERLEY V. L., SCOLLO A., GOTTARDO F., MALCOLM E. M., EDWARDS S. A., LEACH M. C.: The assessment of facial expressions in piglets undergoing tail docking and castration: toward the development of the piglet grimace scale. *Frontiers in veterinary science 3* (2016). 1

[Jir14] JIRKOF P.: Burrowing and nest building behavior as indicators of well-being in mice. *Journal of neuroscience methods 234* (2014), 139–146. 1

[KEH*18] KOPACZKA M., ERNST L., HECKELMANN J., SCHORN C., TOLBA R., MERHOF D.: Automatic key frame extraction from videos for efficient mouse pain scoring. In *5th International Conference on Signal Processing and Integrated Networks (SPIN)* (2018). 2

[Kri09] KRIZHEVSKY A.: Learning multiple layers of features from tiny images. 4

[KSH*07] KUNZ M., SCHARMANN S., HEMMETER U., SCHEPELMANN K., LAUTENBACHER S.: The facial expression of pain in patients with dementia. *PAIN® 133*, 1 (2007), 221–228. 1

[KTFL12] KEATING S. C., THOMAS A. A., FLECKNELL P. A., LEACH M. C.: Evaluation of EMLA cream for preventing pain during tattooing of rabbits: changes in physiological, behavioural and facial expression responses. *PloS one 7*, 9 (2012), e44437. 1

[LBC*10] LANGFORD D. J., BAILEY A. L., CHANDA M. L., CLARKE S. E., DRUMMOND T. E., ECHOLS S., GLICK S., INGRAO J., KLASSEN-ROSS T., LACROIX-FRALISH M. L., ET AL.: Coding of facial expressions of pain in the laboratory mouse. *Nature methods 7*, 6 (2010), 447–449. 1, 2

[LMR17] LU Y., MAHMOUD M., ROBINSON P.: Estimating sheep pain level using facial action unit detection. In *Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on* (2017), IEEE, pp. 394–399. 1, 2

[ML15] MILLER A. L., LEACH M. C.: The mouse grimace scale: a clinically useful tool? *PLoS One 10*, 9 (2015), e0136000. 1

[MRC*16] MCLENNAN K. M., REBELO C. J., CORKE M. J., HOLMES M. A., LEACH M. C., CONSTANTINO-CASAS F.: Development of a facial expression scale using footrot and mastitis as models of pain in sheep. *Applied Animal Behaviour Science 176* (2016), 19–26. 1

[PGC*17] PASZKE A., GROSS S., CHINTALA S., CHANAN G., YANG E., DEVITO Z., LIN Z., DESMAISON A., ANTIGA L., LERER A.: Automatic differentiation in PyTorch. 3

[Prk09] PRKACHIN K. M.: Assessing pain by facial expression: facial expression as nexus. *Pain Research and Management 14*, 1 (2009), 53–58. 1

[RB59] RUSSELL W. M. S., BURCH R. L.: The principles of humane experimental technique. 1

[RFB15] RONNEBERGER O., FISCHER P., BROX T.: U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (2015), Springer, pp. 234–241. 2

[RSP*17] REIJGWART M. L., SCHOEMAKER N. J., PASCUZZO R., LEACH M. C., STODEL M., DE NIES L., HENDRIKSEN C. F., VAN DER MEER M., VINKE C. M., VAN ZEELAND Y. R.: The composition and initial evaluation of a grimace scale in ferrets after surgical implantation of a telemetry probe. *PloS one 12*, 11 (2017), e0187986. 1

[SSZ*11] SOTOCINAL S. G., SORGE R. E., ZALOUM A., TUTTLE A. H., MARTIN L. J., WIESKOPF J. S., MAPPLEBECK J. C., WEI P., ZHAN S., ZHANG S., ET AL.: The rat grimace scale: a partially automated method for quantifying pain in the laboratory rat via facial expressions. *Molecular pain 7*, 1 (2011), 55. 1, 2

[TMJ*18] TUTTLE A. H., MOLINARO M. J., JETHWA J. F., SOTOCINAL S. G., PRIETO J. C., STYNER M. A., MOGIL J. S., ZYLKA M. J.: A deep neural network to assess spontaneous pain from mouse facial expressions. *Molecular pain 14* (2018), 1744806918763658. 1

[TTK14] TAPPE-THEODOR A., KUNER R.: Studying ongoing and spontaneous pain in rodents–challenges and opportunities. *European Journal of Neuroscience 39*, 11 (2014), 1881–1890. 1

[WH14] WHITTAKER A. L., HOWARTH G. S.: Use of spontaneous behaviour measures to assess pain in laboratory rats and mice: How are we progressing? *Applied Animal Behaviour Science 151* (2014), 1–12. 1