computers

MDPI

Article

Markerless Dog Pose Recognition in the Wild Using ResNet Deep Learning Model

Srinivasan Raman [1], Rytis Maskeliūnas [1,*] and Robertas Damaševičius [2]

[1] Department of Multimedia Engineering, Kaunas University of Technology, 51423 Kaunas, Lithuania; Srinivasan.Raman@ktu.edu
[2] Department of Applied Informatics, Vytautas Magnus University, 44404 Kaunas, Lithuania; robertas.damasevicius@vdu.lt
* Correspondence: rytis.maskeliunas@ktu.lt

Abstract: The analysis and perception of behavior has usually been a crucial task for researchers. The goal of this paper is to address the problem of recognition of animal poses, which has numerous applications in zoology, ecology, biology, and entertainment. We propose a methodology to recognize dog poses. The methodology includes the extraction of frames for labeling from videos and deep convolutional neural network (CNN) training for pose recognition. We employ a semi-supervised deep learning model of reinforcement. During training, we used a combination of restricted labeled data and a large amount of unlabeled data. Sequential CNN is also used for feature localization and to find the canine's motions and posture for spatio-temporal analysis. To detect the canine's features, we employ image frames to locate the annotations and estimate the dog posture. As a result of this process, we avoid starting from scratch with the feature model and reduce the need for a large dataset. We present the results of experiments on a dataset of more than 5000 images of dogs in different poses. We demonstrated the effectiveness of the proposed methodology for images of canine animals in various poses and behavior. The methodology implemented as a mobile app that can be used for animal tracking.

Keywords: dog pose recognition; markerless pose estimation; animal tracking; animal behavior analysis; deep learning

Citation: Raman, S.; Maskeliūnas, R.; Damaševičius, R. Markerless Dog Pose Recognition in the Wild Using ResNet Deep Learning Model. *Computers* **2022**, *11*, 2. https://doi.org/10.3390/computers11010002

Academic Editor: Wenbing Zhao

Received: 4 November 2021
Accepted: 22 December 2021
Published: 24 December 2021

1. Introduction

In the area of neuroscience, the analysis and perception of behavior has usually been a crucial task for researchers. Various methodologies such as recording of animal behavior in numerous settings provide smooth methods for annotation yet observing the precise aspects of a behavior is extremely time-consuming [1]. Some results have shown that manual marking on canine features to study reference pointers with the use of software consequences distresses behavior evaluation. A solution has been proposed to rectify the above concerns by introducing the usage of numerous depth sensors on unmarked 3-dimensional (3D) motion capture systems [2], optical tracking [3], and, estimation of dog posture using deep neural network (DNN) [4]. However, previous methods can only be used in limited and controlled environments.

Modern studies have focused on a deep learning methodology [5] to alleviate the usage of classical hand-made image features in feature engineering and digital image processing and to develop user-defined tracking on different kinds of animal, where we can avoid the usage of large data or training models from scratch. This methodology is based on receiving pre-trained weights from a deep learning model and applying transfer learning [6]. This is a learning approach that generalizes between the base and target domains and delivers variant distributions. The current work on DNNs reveals that transfer learning has convenient features that were generalized well to similar pioneering tasks. As profound highlights in the end change from general to specific along the system, include

transferability drops significantly in higher undertaking specific layers with expanding area disparity. Deep learning achieves extremely reliable overall efficiency on numerous pose detection benchmarks, but to improve this performance, the deep learning model is trained on a large collection of categorized animal photographs. Multiple frames are marked in the canine feature outlines to demonstrate that this methodology can be efficiently proposed for animal behavior analysis.

YOLO [7] is a real-time object detection methodology that has practically proved practically that the speed and accuracy of object detection is extremely high. This methodology does not need to retrain the dataset. Instead, we can change the size of the image. In the previous supervised learning method, the training data localizations are reused to detect the feature posture by applying numerous frames at multiple locations on the model. These methodology detections are found by locating the multiple regions scored on the image, but it is time-consuming since the data must be retrained. To avoid these limitations, the YOLO model uses an entirely different approach by applying a single neural network completely on the image, which is further divided into multiple regions, and hence the detection is based on probability of bounding boxes of weighted predictions. Domain adaptation evidently describes that there is a necessity of enabling the trained datasets manually on the model for object detections under significant variations in posture and other labels. The COCO data set [8] is used with the YOLO technique to simplify object segmentation and labelling. The COCO data set is designed for object detection, segmentation, localization detection, and label generation on a large image dataset. A significant strategy for domain adaptation is to understand the domain-invariant models between the source datasets and the target dataset via a latent isomorphism-inducing latent on different domains.

Deep lab cut (DLC) [9,10] is a method for 3D markerless pose estimation based on transfer learning with deep convolutional networks that combines algorithms for object detection and semantic image segmentation (pre-trained deep neural ResNet models and CNN layers). DLC can accurately convert large videos into low-dimensional time-order data with semantic denotation. This CNN layer is used to sample-up the image/video information to produce spatial probability weights instead of classifying the layer at the output of the ResNet. The probability weight for each characteristic of the canine's body denotes the 'evidence' that the characteristic of the canine's body is in a specific setting (that is, pose). A DLC methodology can be used to derive distributed representations through domain adaptation of transfer learning [11].

The recognition of pose of animals discussed in this paper is used to segment the pose of dogs. Animals commonly have a diverse range of variations on poses and there is no available canine pose dataset for model training and testing. To address this problem, we constructed an animal pose dataset to facilitate model training and evaluation. With respect to the heavy effort involved in labelling the dataset, and given that it is difficult to label data for all concerned canine types, a proposed method of a different traverse-domain variation methodology for converting the canine pose considerate from classified animal instructions to non-labelled animal trainings is introduced. In this paper, our aim is to tackle the problem of recognition of animal poses, which has numerous applications in ecology, zoology, and entertainment.

Previous research [12,13] has focused only on human pose recognition or gesture recognition [14] and produced promising results. Animal posture recognition is part of a larger domain of animal behavior studies, which also includes voice analysis [15]. Image segmentation is an important part of various computer vision tasks such as setting environment scene recognition [16] and route planning and finding [17]. In supervised learning, it is a known fact that there is only a limited chance to simply train the training sets on the model for each new domain. Hence, we need an algorithm that utilizes limited labeled data across multiple domains.

Key problems therefore can be identified as lack of adaptation of human related pose studies to the animal domain as well as lack of any categorized data of the animal domain. Our contributions are as follows:

1. We used the technique for markerless pose recognition based on an open-source Deep Lab Cut learning tool set (DLC). The location of the data improves the robustness and provides higher performance in the outside domain where it yields an accurate result due to the training of the data from scratch.
2. We add a novel version of networks to the DL package, MobileNetV2s, which pave the way for the most accurate and fast posture detection of the animal.
3. We train the DNN to classify 18 image features and excerpt the canine posture by detecting a set of features. This methodology has been experimented with more than 5000 manually marked canine photographs.
4. We examine the performance of the proposed methodology for dog posture tracking in various settings and offer an openly available toolbox for the research community.

Our paper comprises five sections. Section 2 discusses the existing works of various authors and our innovative contribution to defining the process and its functionalities. Section 3 describes our methodology and the models used. Section 4 explains the experimental study and the results. Section 5 describes the conclusion of this study and future work.

2. Related Works

This section discusses previous work in the domain of animal tracking techniques using computer vision systems for animal posture recognition in general, and dog posture recognition in particular. In current years, from the large surveillance cameras installed in the natural environments, a few researchers have made attempts to recognize animals through the usage of computer vision methods for some of species, including African penguins, northeast dogs, lemurs, first rate white sharks, primates, ringed seals, large pandas, and crimson pandas. They extracted discriminative features from positive body parts of animals and differentiated poses based on the extracted functions. Similar strategies were implemented for cattle, dairy cows, and pigs in agricultural applications.

Pose recognition focuses on predicting body joints on detected objects/mostly humans. However, skeleton detection on animals is rarely studied and faces many challenges [18]. The notable exceptions are Anipose [19], which allows 3D markerless tracking of animal skeletons, LEAP [4] to track the full animal pose, and RGBD-Dog [20] which specifically designed for tracking dog skeleton data.

For example, Alameer et al. [21] adopted you-only-look-once (YOLO) and faster regions with CNN features (Faster R-CNN) with deep residual network (ResNet-50) to recognize the postures of pigs, i.e., standing, sitting, lying lateral, and lying sternal. Ayadi et al. [22] used a standard VGG16 model to recognize cow postures, specifically rumination behaviour, from video recordings. Brünger et al. [23] used a U-Net network with different encoder architectures for panoptic pig segmentation, which is a combination of semantic segmentation (assigning a class label to each pixel) and instance segmentation (detecting and segmenting each object instance), while the results are used for posture detection. Hahn-Klimroth et al. [24] presented a multistep CNN system to detect three typical African ungulate stances in zoo enclosures, including model averaging and postprocessing rules to make the system robust to outliers. Liu et al. [25] used a ResNet backbone, three transposed convolution layers, and a final output layer to estimate the pose. The model is evaluated using data sets from four different animal species (mouse, fruit fly, zebrafish, and monkey).

Shao et al. [26] investigated an automatic method for recognizing the posture states of pigs. To determine the effect of posture monitoring using Resnet, Xception, and MobileNet networks, they could acquire key frames from the image, detect a single pig in each frame, extract the contours of each animal, and distinguish their posture. Wang et al. [27] developed a dog motion attitude fusion method, which can capture different posture data of the police dog, including standing, sitting, lying, etc. Wang et al. [28] investigated how deep learning can be used to detect and categorize lameness in horses. They presented a markerless approach using DeepLabCut that uses ResNet-50 to perform deep learning on hundreds of images labeled with user-labeled horse body parts. They discovered that

the trained model could accurately identify horse body parts and determine whether a horse was lame. Wu et al. [29] proposed Deep Graph Pose (DGP), a probabilistic graphical model built on deep neural networks, to exploit useful temporal and spatial constraints, and structured variational approach is developed to perform inference in this model. The developed semi-supervised model makes use of both labeled and unlabeled frames to achieve accurate tracking of animal poses. Finally, Zhang et al. [30] detected multiple animals (kangaroos, emu, dingos, birds, and wildcats) in the wild in an omni-supervised learning setting using two CNN based detectors (Faster R-CNN, SSD) and two CNN based classifiers (Inception V3 and MobileNet).

To summarize, despite the incredible precision attained in these investigations, they largely demand the animals to be photographed under well-controlled settings. However, such an assumption is frequently impractical for recognizing animals in the field due to large variances in the animals' body poses and the lighting of the images. In this research, a unique end-to-end approach is suggested to learn different capacities for estimating dog poses in the field. The suggested approach is not dependent on posture recognition modules and can deal with large pose changes more accurately by introducing an auxiliary goal of simplified pose categorization to drive feature learning.

3. Methods

3.1. Pose Recognition Network Model

The main aim was to train the DNN to classify 18 image features and extract the canine posture by detecting a set of features. The input data was specified as RGB and DEPTH images as captured by a stereo camera feed. The output data (logical combination of keypoints, e.g., of a sitting dog) is a classified 3D pose of a canine.

Our pose recognition is based completely on some convolutional layers delivered on a ResNet [31] in this work. The main objective of the DNN model is to provide an accurate output for the given input using the weights that are associated with each focal point on the image. Here, we use this DNN to generate the training data for the estimation of the posture of the dog. We use two main class of models, MobileNetV2 [32] and ResNet [33], to build a very convenient framework for our model. ImageNet is produced by combining canine dataset with another larger dataset with images and training a CNN on these merged examples. Our model uses two sets of canine images; one set is based on trained dataset, and another is without training the data sets. The comparative results in using the pre-trained model on ImageNet instead of training the datasets from scratch improves performance and provides the best result on the canine posture estimation.

The CNN model receives an RGB image and depth data as input; this part of the process is known as 'Depth-aware CNN'. The analyzed result is passed to the next phase of finding the regressed coordinates to plot the image as dots from the input. The final phase is the estimation of posture of the input image and is known as the geometric pose recognition phase wherein CNN model is applied and the output 3-Dimentional pose of the input image is derived (Figure 1).

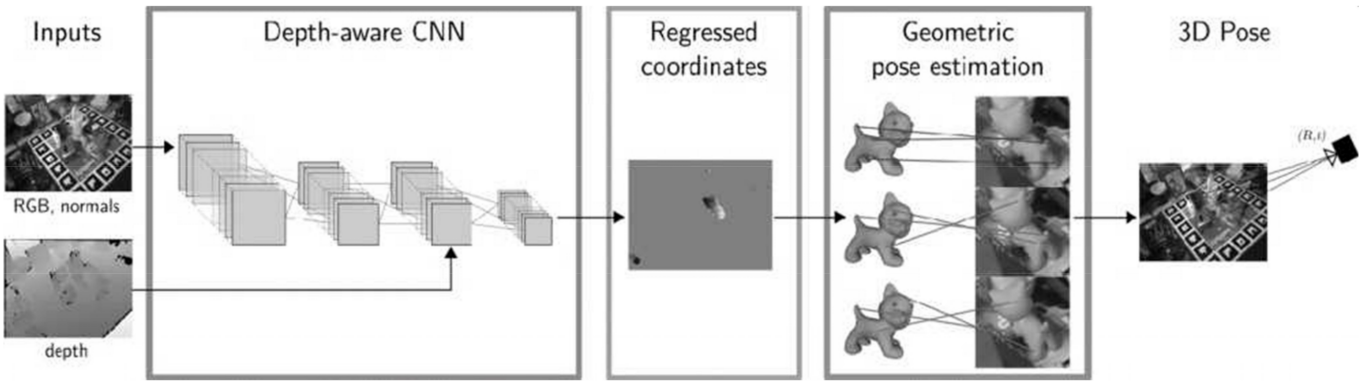

Figure 1. Pose recognition using CNN-3D.

3.2. Feature Extraction

Recognition and detection of human poses are very widely used in neural networks, as they have excellent accuracy and effectiveness on larger datasets [34,35]. However, there is a limitation in the DNN model since the minute intersections or joints of a canine feature detection are very confused to detect the pose. Many neural networks face challenges in observing the background environment during implementation, and these challenges result in a failure to depict the exact pose of model. Furthermore, the reason for this failure can also be the very rare posture capture of human being such as diving and skiing, which are an additional bottleneck in detecting the pose. Likewise, animals also face an extra challenge in detecting their poses, since they require more training datasets to run the model on different frames. Therefore, the detection of canine poses from the annotated trained datasets is precise for generalization systematically.

This feature extraction process involves training a DNN in protected posture estimation and posing ResNet techniques on the preset ImageNet locations of the features (Figure 2). The trained model was then passed over to deconvolutional layers in order to characterize the region corrections are required.

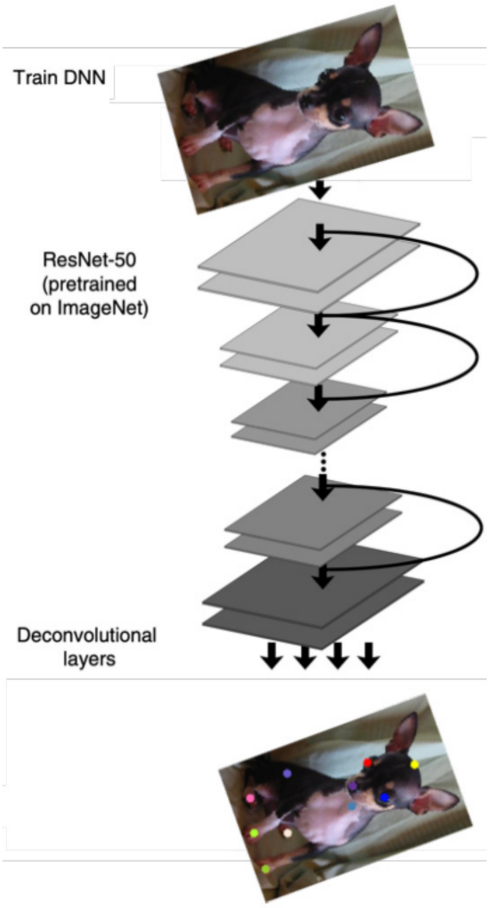

Figure 2. Feature extraction for marker recognition.

3.3. Framework of Dog Pose Estimation

The Deep Residual Neural Network (ResNet [33]), a deep learning architecture, and preliminary training weights are utilized by the DLC tool set. Specifically, as shown in Figure 3, the parameters are illustrated to adapt to this toolkit by applying a 50-layer ResNet configuration. This DLC model takes an input of some dimensional RGB image (we use 640 × 640) RGB image along with the characteristic and posture of the canine with some number of frames per image. Here, we used 17 frames per image. Hence, for each body feature of canine in the input image, the output will be 640 × 640× number of frames score

chart for each pixel with activation values. In DLC, we receive input with filters (7×7), the batch model, and the pool (Max 3×3) in a CNN which is a deep neural network that is frequently used to detect the visual image. Two sets of blocks, namely identity blocks and convolution blocks, are used to parse the datasets hence each of this block CB, IB contains 3 CNN layers, a dataset models, and a rectified linear activation unit, or ReLU for short, is added. After certain trainings on the datasets, they are further compressed in the deconvolution layer along with the average pool.

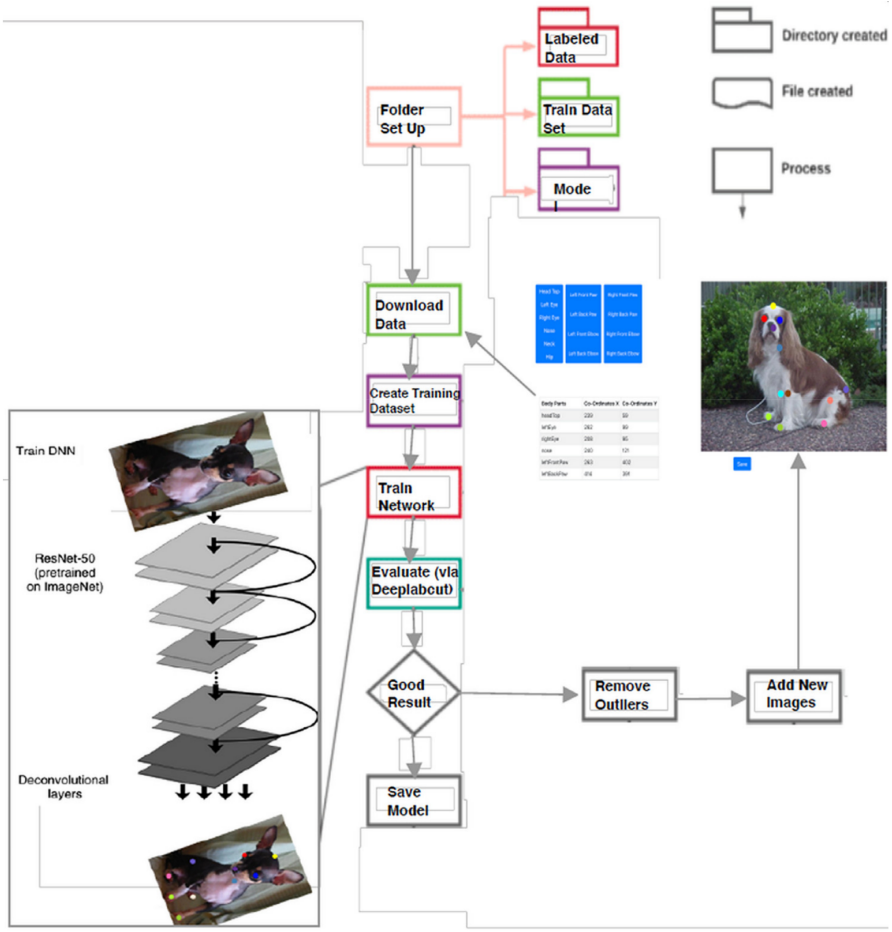

Figure 3. Workflow of our proposed methodology for dog pose estimation.

We have used the principles of neural architecture search (NAS) for tuning the parameters and adopted the autokeras library in Python. By constructing a neural network kernel and a method for optimizing acquisition function in tree-structured space, it enables Bayesian optimization to direct the search. The best performance was achieved using the following values of the hyperparameters: optimization function, ADAM; batch size is 64; learning rate is $10-3$; weight's initialization is Xavier; bias initialization is equal to zero; and early stopping criteria is 30 epochs.

3.4. Canine Image Recognition

We have applied DLC and trained the model on the training dataset of thousand pictures with suitable annotation. After the training process of datasets, this DLC principle is found to be capable of appropriately analyzing and finding dog body parts.

The YOLO network creates a map by these prediction bounding boxes of all of the objects in the image and a bunch of boxes ranked by their confidence value. This can help us to find where the objects are in the image. Now we can predict the detected object by predicting the class probability of each.

- Batch normalization was performed after each layer. This helped increase mAP to almost 2%. Thus, the dropout layer that was used earlier was no longer required.
- To further increase detection accuracy, the images were resized to a high resolution of (448 × 448). Therefore, increase the mAP to 4%.
- A non-maximum suppression method is implemented on the obtained results to refine the output. For each class, the low-probability predictions are discarded. This is carried out by setting a threshold value. Predictions below the threshold are suppressed, and thus the final output has only features that are detected with a higher confidence value.

Anchor boxes were used to predict the bounding boxes. If the input for YOLO is an RGB image with a resolution of resolution 608 × 608, the output will have the recognized canines in an image. For the very 19 × 19 cell, the maximum probability score is calculated by taking the mean across five anchor boxes and across different classes. The cells are colored to an object that is most likely to be in a better way for understanding. Figure 4 shows an example of what the bounding boxes of recognized canines in the images will look like.

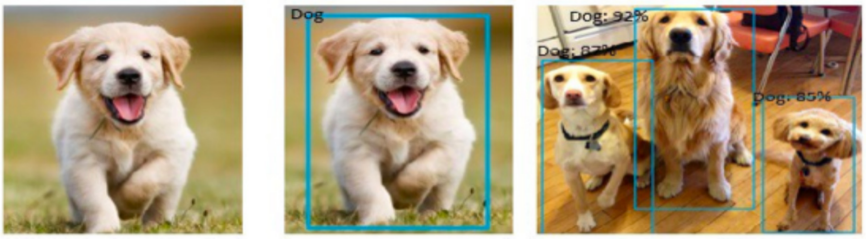

Figure 4. Color grid for dog identification.

4. Experiments and Results

4.1. Data Set and Experimental Setting

Our dataset is gathered and categorized with our canine mammal model, which consists of 5000 RGB (Red, Green, Blue) images of a canine dog with multiple postures and in various environments. Initially, we divided our dataset into three subsets. We used a training data set of ~76% images; for data set validation we used ~4% images; and finally we used ~20% images for testing. The test dataset was collected manually from royalty-free videos and images (5000 images in total) and then manually labeled (see Section 4.2 step 2).

To train the model, each image has manually annotated features data. Pixel points with higher annotations detected on the canine feature are predicted as the model position. In addition to model prediction, the derived datasets are trained for 1000 to 2000 epochs using a deep learning model with Tensorflow support, which uses Python language at its backend. The pre-trained weights are derived using the ResNet v1_50 checkpoint file.

We use GeForce® GTX 1080 Ti is NVIDIA's new leading gaming GPU, based on the NVIDIA Pascal™ architecture for training the dataset with time lateral of 17 min/10,000 iterations to provide the best outcome.

4.2. Procedure for Using the DLC Toolbox

We follow the following procedure:

1. Training the data set: Collect images with different positions features of the canine animal behavior annotations and select the minimum region-of-interest (ROI), and hence while annotation is compressed the sample dataset is attained.
2. Manual labelling: Locate the different body characteristics of the dog: The intersection parts of the canine feature are marked manually (for example, the wrist and elbow are marked as features of interest).
3. Train a DNN architecture using the DLC toolbox: The manually labeled canine feature is further trained using DNN to predict body part location using the base image loaded in the framework. A different information layer is derived within a part of canine feature to predict the possibility that a canine body feature is in a specific pixel.

Training of data adjusts both information and DNN weights and further storage. The trained network (DNN) can be used to derive the positions of canine body parts from images or videos. The images show the most likely locations of canine body parts for 17 labeled canine body parts.

4.3. Markerless Dog Body Annotation

The trained models derived from the provided DNN architecture are used on unmarked images and videos. The canine features are located using the DLC toolbox. Therefore, the estimation of the dog's posture is based on the detected characteristic and is represented using the color of each characteristic of different postures and environments. The canine features are connected by line segments using the color as follows: blue represents the head; red represents the legs of dog; yellow represents the paws, and green represents the torso region. Figure 5 shows the manually marked canine feature in different postures.

Figure 5. Dog with marked annotations: blue represents the head; red represents the back legs of the dog; cyan represents the frontal legs; and green represents the torso region.

The illustration of markerless dog body part annotation using our developed application are presented in Figure 6.

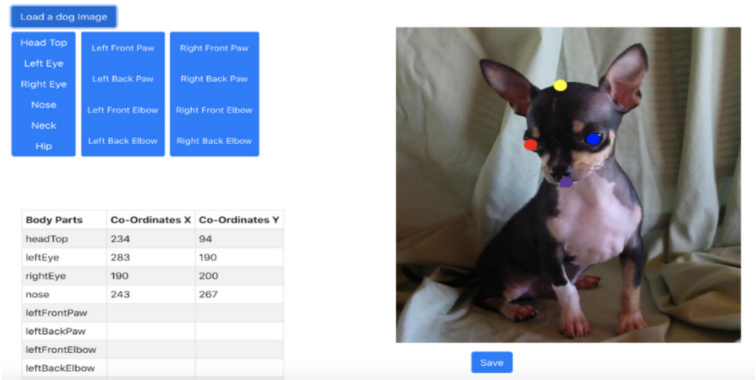

Figure 6. Dog body part label annotation.

4.4. Results

A CNN model is used to study the indications of the characteristics by comparing the performance of the different postures and body parts of the dog for our proposed model. The model is used for comparison by models with 2000+ images for training and further validated on 550+ images. The resolution of a single canine is predicted to be 640 × 640 RGB image format. To evaluate the canine feature identification, we implemented a cross-validation between the base model and the target model, and hence as the result of comparison the pixel error is noted. While the model is evaluated, it is predicted that the pixel error for testing data and trained datasets are decreasing subsequent when it

reaches 100,000 training iterations. Hence, the iteration is minimized to 100,000 iterations for further analysis, thereby considering the balance between training speed and time taken for training datasets.

For evaluation, we used displacement between the accurate truth in the datasets and the future model, which is called the pixel error (or root mean squared error, RMSE). The results in validation loss of 0.089 are achieved in by the CNN model, whereas the training loss on validation of 0.0162 is achieved. Furthermore, as shown in Table 1, compared to ground truth, the model possesses less RMSE pixels of the dog's predicted feature location, thus verifying that our framework is very reliable in detecting each feature, resulting in approximately 25+ pixel units of errors in the test dataset. Our proposed model has the capacity to locate more features of the canine with great assurance along with video frames. After several iterations, it is found that most of the key joints, especially the dog nose, are observed to maintain high assurance on prediction approximately to 1 pixel on each frame in the dataset model. Locus points marked near the ankles and joints are hard to predict, as they have a very low detection point due to the similarity in the canine body. The confusion exists due to the similar features of the dog body part, especially near the ankles and wrists. Table 2 presents the calculated match rate [36].

Table 1. RMSE values (px) of dog features.

Dog Feature	Training Image Dataset	Testing Image Dataset
Eye Left	2.81	4.10
Eye Right	2.85	4.41
Nose	2.79	5.44
Head Top	3.62	9.01
Hip	3.67	11.28
Right Back Paw	5.77	19.87
Right Front Paw	4.21	22.18
Left Back Paw	3.64	21.35
Left Front Paw	4.16	9.35
Left Front Elbow	6.78	21.68
Right Front Elbow	4.09	23.63
Left Back Elbow	4.16	18.33
Right Back Elbow	2.94	24.64
Neck	1.94	21.83
Overall Detection	**2.91**	**20.12**

Table 2. Correct match rate of dog features.

Dog Feature	Training Image Dataset	Testing Image Dataset
Eye Left	0.9539	0.9434
Eye Right	0.9536	0.9409
Nose	0.9541	0.9325
Head Top	0.9473	0.9034
Hip	0.9469	0.8849
Right Back Paw	0.9298	0.8149
Right Front Paw	0.9425	0.7960
Left Back Paw	0.9471	0.8028
Left Front Paw	0.9429	0.9006
Left Front Elbow	0.9216	0.8001
Right Front Elbow	0.9435	0.7842
Left Back Elbow	0.9429	0.8274
Right Back Elbow	0.9529	0.7760
Neck	0.9610	0.7989
Mean	**0.9462**	**0.8479**

5. Discussion and Conclusions

State-of-the-art methods extract local functions from unique image frame parts of dogs based totally on stand-alone pose recognition techniques. As a result, they are constrained by the pose recognition accuracy and suffer from self-occluded body elements. Instead of estimating complex body poses, this study simplifies canine poses and uses this information as a pose classification task to oversee characteristic learning.

In this paper, we use a semi-supervised reinforcement deep learning model for dog pose recognition. Rather than fully using the deep learning model as other authors have carried out, we used a combination of limited labeled data and a large amount of unlabeled data during the training of datasets. The trained data are derived using the YOLO model on the COCO dataset. Sequential CNN is also implemented for feature localization and to locate the actions and posture of the canine to provide spatio-temporal analysis. We use image frames to locate the canine feature and locate the annotations. To run a spatial-temporal analysis, we have implemented a combined model on this feature to calculate the dog posture. The proposed model based on ResNet deep learning model learns numerous corresponding capabilities by using steering extraordinary characteristic extraction network branches in the direction of exceptional regions of the canine body through erasing activated regions from enter canine images. By fusing the pose-guided complementary capabilities, this paper successfully improves the canine re-identification accuracy, as demonstrated within the assessment experiments on a benchmark dataset. As a result of this methodology, we can avoid training the feature model from scratch and minimize the need for a large dataset.

By adaptively erasing partial regions on canine photographs, our method can force one-of-a-kind feature extraction branches to recognize distinct elements of canine images. Extensive assessment experiments on our self-collected dataset shows that our proposed approach can appreciably improve the accuracy of canine re-identification. The strengths of the method can be considered its unique adaptation in the domain of animals (canine) and its ability to categorize the poses of dogs. The main limitations are related to the dynamic nature of an animal–in real life the animal constantly moves, even if performing some trained action (such as the command 'sit'), creating fluctuations in stereo camera feed and affecting accuracy. Future work of this algorithm will consider adapting feed filtering and stabilization methods to pass the more stable data to the classification backend. Furthermore, the algorithm itself will be trained with such unstable data as we will collect more and more of it. In the future, we are going to further expand our proposed technique to different species.

Author Contributions: Conceptualization, R.M.; methodology, R.M.; software, S.R.; validation, R.M. and R.D.; formal analysis, R.M. and R.D.; investigation, S.R. and R.M.; resources, R.M.; data curation, S.R.; writing—original draft preparation, S.R. and R.M.; writing—review and editing, R.D.; visualization, S.R. and R.M.; supervision, R.M.; funding acquisition, R.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data has been presented in main text and the dataset is available online at: http://dlib.net/files/data/ (accessed on 4 November 2021) and https://dogo.app/ (accessed on 4 November 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Akihiro, N.; Ishida, Y.; Uezono, N.; Funaya, H.; Nakashima, K.; Suzuki, T.; Shibata, T.; Wakana, S. Low-cost three-dimensional gait analysis system for mice with an infrared depth sensor. *Neurosci. Res.* **2015**, *100*, 55–62.
2. Nakamura, T.; Hori, E.; Matsumoto, J.; Bretas, R.V.; Takamura, Y.; Ono, T.; Nishijo, H. A markerless 3D computerized motion capture system incorporating a skeleton model for monkeys. *PLoS ONE* **2016**, *11*, e016615411. [CrossRef] [PubMed]
3. Nashaat, M.A.; Oraby, H.; Peña, L.B.; Dominiak, S.; Larkum, M.E.; Sachdev, R.N. Pixying Behavior: A Versatile Real-Time and Post Hoc Automated Optical Tracking Method for Freely Moving and Head Fixed Animals. *eNeuro* **2017**, *4*. [CrossRef] [PubMed]
4. Pereira, T.D.; Aldarondo, D.E.; Willmore, L.; Kislin, M.; Wang, S.S.; Murthy, M.; Shaevitz, J.W. Fast animal pose estimation using deep neural networks. *Nat. Methods* **2019**, *16*, 117–125. [CrossRef]
5. Mathis, A.; Mamidanna, P.; Abe, T.; Cury, K.M.; Murthy, V.N.; Mathis, M.W.; Bethge, M. Markerless tracking of user-defined features with deep learning. *arXiv* **2018**, arXiv:1804.03142.
6. Zagoruyko, S.; Komodakis, N. Wide residual networks. *arXiv* **2016**, arXiv:1605.07146.
7. Yang, W.; Jiachun, Z. Real-time face detection based on YOLO. In Proceedings of the 1st IEEE International Conference on Knowledge Innovation and Invention, Jeju, Korea, 23–27 July 2018.
8. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Contex. Springer: Cham, Switzerland, 2018.
9. Pishchulin, L.; Tang, S.; Andres, B.; Insafutdinov, E.; Andriluka, M.; Gehler, P.V.; Schiele, B. Deepercut: Joint subset partition and labeling for multi person posture estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4929–4937.
10. Insafutdinov, E.; Pishchulin, L.; Andres, B.; Andriluka, M.; Schiele, B. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 34–50.
11. Mathis, A.; Biasi, T.; Schneider, S.; Yuksekgonul, M.; Rogers, B.; Bethge, M.; Mathis, M.W. Pretraining boosts out-of-domain robustness for pose estimation. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), Seattle, WA, USA, 3–8 January 2021. [CrossRef]
12. Kulikajevas, A.; Maskeliunas, R.; Damaševičius, R. Detection of sitting posture using hierarchical image composition and deep learning. *PeerJ Comput. Sci.* **2021**, *7*, e447. [CrossRef]
13. Žemgulys, J.; Raudonis, V.; Maskeliūnas, R.; Damaševičius, R. Recognition of basketball referee signals from real-time videos. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *11*, 979–991. [CrossRef]
14. Mujahid, A.; Awan, M.J.; Yasin, A.; Mohammed, M.A.; Damaševičius, R.; Maskeliūnas, R.; Abdulkareem, K.H. Real-time hand gesture recognition based on deep learning YOLOv3 model. *Appl. Sci.* **2021**, *11*, 4164. [CrossRef]
15. Maskeliunas, R.; Raudonis, V.; Damasevicius, R. Recognition of emotional vocalizations of canine. *Acta Acust. United Acust.* **2018**, *104*, 304–314. [CrossRef]
16. Petraitis, T.; Maskeliūnas, R.; Damaševičius, R.; Połap, D.; Woźniak, M.; Gabryel, M. Environment scene classification based on images using bag-of-words. *Stud. Comput. Intell.* **2019**, *829*, 281–303. [CrossRef]
17. Malūkas, U.; Maskeliūnas, R.; Damaševičius, R.; Woźniak, M. Real time path finding for assisted living using deep learning. *J. Univers. Comput. Sci.* **2018**, *24*, 475–487.
18. Cao, J.; Tang, H.; Fang, H.; Shen, X.; Lu, C.; Tai, Y. Cross-domain adaptation for animal pose estimation. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 9497–9506. [CrossRef]
19. Karashchuk, P.; Rupp, K.L.; Dickinson, E.S.; Walling-Bell, S.; Sanders, E.; Azim, E.; Tuthill, J.C. Anipose: A toolkit for robust markerless 3D pose estimation. *Cell Rep.* **2021**, *36*, 109730. [CrossRef]
20. Kearney, S.; Li, W.; Parsons, M.; Kim, K.I.; Cosker, D. RGBD-dog: Predicting canine pose from RGBD sensors. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 8333–8342. [CrossRef]
21. Alameer, A.; Kyriazakis, I.; Bacardit, J. Automated recognition of postures and drinking behaviour for the detection of compromised health in pigs. *Sci. Rep.* **2020**, *10*, 13665. [CrossRef]
22. Ayadi, S.; Ben Said, A.; Jabbar, R.; Aloulou, C.; Chabbouh, A.; Achballah, A.B. Dairy cow rumination detection: A deep learning approach. In *International Workshop on Distributed Computing for Emerging Smart Networks*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 123–139. [CrossRef]
23. Brünger, J.; Gentz, M.; Traulsen, I.; Koch, R. Panoptic segmentation of individual pigs for posture recognition. *Sensors* **2020**, *20*, 3710. [CrossRef]
24. Hahn-Klimroth, M.; Kapetanopoulos, T.; Gübert, J.; Dierkes, P.W. Deep learning-based pose estimation for african ungulates in zoos. *Ecol. Evol.* **2021**, *11*, 6015–6032. [CrossRef]
25. Liu, X.; Yu, S.; Flierman, N.A.; Loyola, S.; Kamermans, M.; Hoogland, T.M.; De Zeeuw, C.I. OptiFlex: Multi-frame animal pose estimation combining deep learning with optical flow. *Front. Cell. Neurosci.* **2021**, *15*, 621252. [CrossRef]
26. Shao, H.; Pu, J.; Mu, J. Pig-posture recognition based on computer vision: Dataset and exploration. *Animals* **2021**, *11*, 1295. [CrossRef]
27. Wang, Y.; Huang, Q.; Chen, S.; Zhu, C. From state estimation for dogs to the internet of dogs. In Proceedings of the 2019 IEEE 4th International Conference on Image, Vision and Computing ICIVC, Xiamen, China, 5–7 July 2019; pp. 748–753. [CrossRef]

28. Wang, Y.; Li, J.; Zhang, Y.; Sinnott, R.O. Identifying lameness in horses through deep learning. In Proceedings of the ACM Symposium on Applied Computing, Virtual Event, Korea, 22–26 March 2021; pp. 976–985. [CrossRef]
29. Wu, A.; Kelly Buchanan, E.; Whiteway, M.R.; Schartner, M.; Meijer, G.; Noel, J.; Paninski, L. Deep graph pose: A semi-supervised deep graphical model for improved animal pose tracking. bioRxiv; 2020. [CrossRef]
30. Zhang, T.; Liu, L.; Zhao, K.; Wiliem, A.; Hemson, G.; Lovell, B. Omni-supervised joint detection and pose estimation for wild animals. *Pattern Recognit. Lett.* **2020**, *132*, 84–90. [CrossRef]
31. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. *arXiv* **2015**, arXiv:1512.03385.
32. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; IEEE: Miami, FL, USA. [CrossRef]
33. Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; Li, F.F. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops), Miami, FL, USA, 20–25 June 2009; IEEE: Miami, FL, USA. [CrossRef]
34. Kulikajevas, A.; Maskeliunas, R.; Damasevicius, R.; Scherer, R. Humannet-a two-tiered deep neural network architecture for self-occluding humanoid pose reconstruction. *Sensors* **2021**, *21*, 3945. [CrossRef]
35. Li, M.; Jiang, Z.; Liu, Y.; Chen, S.; Wozniak, M.; Scherer, R.; Li, Z. Sitsen: Passive sitting posture sensing based on wireless devices. *Int. J. Distrib. Sens. Netw.* **2021**, *17*, 17. [CrossRef]
36. Tharwat, A. Classification assessment methods. *Appl. Comput. Inform.* **2020**, *17*, 168–192. [CrossRef]