Report 16

Samuel Spillane

April 29, 2020

My final update is that I am finishing up the paper. I am attempting to get similar looking diagrams across R and Python by using Seaborne in both languages before I add them to my data section and begin editing. Current the paper is rough and has no editing performed on it of any kind. I will have this finished by the end of today and uploaded to my site, preferably by 5PM. My current unedited progress is as follows:

Currrent Unedited Progress

Nonbinary Classification of Procedural Animation Using Geospatial Analysis and Machine Learning Abstract As the trend emerges in the field of applied machine learning in 3D animation, automatic rigging and animation of characters has become the forefront of the field. Fundamentally, this area of research creates an issue: the lack of identification in the poses of such procedural animations. How does a computer or animator know a character is walking when the animation is not predefined? It is important for other fields such as self-driving cars and videogames to readily know what a character is doing momentarily. To answer this question, this research uses techniques from machine learning and geospatial analysis to classify unlabeled poses. In order to perform such a task, a supervised deep neural network is ultimately discovered as the most accurate way to classify poses of the techniques applied. For context, a discussion regarding the procedural issues with applying geospatial techniques to such a task is also included. Finally, the results, as well as future directions for research, are explored.

Introduction An emergent trend in the field of machine learning is the procedural mathematical modeling of limb positions of humanoid figures using techniques such as generative adversarial or convolutional networks. The purpose of such research has applications in self-driving cars, drone technology, remote or automated surgery, and a variety of other fields. The central goal of this emergent research is to apply a rig or skeleton over a video or photo of a humanoid figure in order to capture the positions and transformations present. For example, an algorithm would have an input of a video of an individual walking and the goal is to get it too automatically rig and animate a 3D model of a character performing the same action. Rigging in this case is the term for applying a virtual skeleton on the 3D model of which the bones each individually control a certain region or limb. Such rigging is often a complex task that involves assigning how much control each bone should have over its region. This process is often expensive and time consuming, therefore automation of such processes is essential. The understanding of where an arm is in three-dimensional space relative to other limbs allows for improved human interaction in tools such as the laparoscopic DaVinci Machine and makes motion capture more economical. []

However, this new research often does not discuss a radical consequence of such methods: when these 3D "skeletons" are applied the ability to discern what the person in question is doing. In other terms, the process of making the animation is automated, but the ability to understand what the subject is doing is lost to a computer. This is important because for self-driving cars, such a technology would allow for cars to better interpret what other drivers are doing by their body language, which is often useful for human drivers. For automated or remote surgery, understanding the actions of surgeon's hands could be useful in the training of highly precise procedures such as laparoscopy. [] Therefore, a novel solution is required for this problem. The primary purpose of this research is to explore different methods from geospatial statistics and machine learning applied to the task of the classification of actions by armature bone positions. The central goal is to find an algorithm that is efficient enough to work in ensemble with networks that automate the rigging and animation process. Such a combination of algorithms would lead to the complete automation of the animation of humanoid characters. For the purposes of this research, three algorithms were applied to the problem: a support vector machine(SVM), a naive Bayes network, and a deep neural network (DNN). These algorithms were chosen because they standards for classification tasks and can also be highly performant. These networks were also chosen because they involve different branches of statistical analysis, namely geospatial analysis, probabilistic reasoning, and curve fitting respectively.

The primary hypothesis of this research is that the positions of the bones are less relevant to classification than their transformations. The basis of this hypothesis is that the position of an individual bone is exclusively derived from its attachment to other bones. For example, a hand bone may be higher or lower, but this is because it is a attached to a rotated parent bone, in this case an arm. In turn, the only way for a bone to be translated independent of connected parent bones is for the entire rig to be moved. A consequence of this

reasoning is that the geospatial technique will be less effective than other methods because the statistical relationship between bone positions is divergent while the correlations between bone transformations is convergent.

Background All of the algorithms attempted for this task are fall under a subfield of machine learning known as supervised classification. In supervised classification, a dataset of labeled parameters or features is given to the algorithm to enable it to learn the factors that define certain classes. In this case the labels for the training data are the names of actions that a character might be doing such as sitting or running. After the features of the classes are learned, data that is not labeled can be classified using the information garnered during the training step. The end result is a model for classifying what a character is doing. This mimics human learning processes of understanding what other individuals are doing. A child learns what a handshake looks like by observing others doing the action.

Support vector machines function as classification algorithms by finding a line of most difference between a set of data points. In order to do this, an SVM takes the labeled training data and finds the equation of a line provided by

STAND FORM LINE EQUATION

that contains the maximal distance to each point. In order to find the line the equation

SVM LAMBDA MAXIMIZATION FORMULA

is applied to each pair of points (x,y). The resulting line divides the classes in space, allowing future unlabeled data points to be easily determined to be either one class or the other. [] Like many geospatial techniques clustering techniques this is non probabilistic, meaning it returns a dichotomized binary yes or no answer to the classification rather than a chance that is the label is accurate. SVMs can be modified to perform probabilistic observations, but this is unnecessary in this task. SVMs can only classify between two labels per line of maximal distance. For the purposes of this research this is not an issue, because the bones are ultimately compared against a neutral poses rather than every other class. This does increase the complexity of using such an algorithm because it entails using one SVM per label with the exception of the neutral pose.

The second approach, naïve Bayes, utilizes Bayes algorithm to classify a data point based of learned differences between classes. Bayes rule,

BAYES RULE

is a mathematical statement of the process of educating guessing. As more information is introduced, the likely that a data point is of a certain class change. For example, given that the data point is question has four legs and a tail, what is the probability that it is dog? The answer is pretty low. However, adding the observation that the data point in question barks increases the likelihood significantly. Similarly, knowing that a leg bones rotation is 90 degrees is applicable in my actions such as sitting and running, but having both leg bones at 90 degrees is only occurrent when a skeleton is sitting. The naïve Bayes algorithm uses this principle to classify data.

The final approach, a deep neural network functions by utilizing the principles of a "perceptron," Which is a mathematical unit designated by, To test this hypothesis a "neutral" pose is included in the data, which in this case is a T-pose. A T-pose refers to a rig standing straight with it's arms positions both a 90 degree angles. The hope is that this neutral pose can be used a reference point for the algorithms to learn the class differences from.

PERCEPTRON

Perceptrons take in a piece of information and weight it using matrix multiplication. The weighted values are then fed into "layers' which are essentially organized matrix operations. In the case of a DNN, the values are repeatedly cross multiplied and weighted again. These values are then normalized. This process is known as forward propagation. With the structure of the network defined, the weights are assigned random values. The training data is then fed into the network and the amount of error, which is called loss, is computing using a standard function. This standard function is derived so that the points of minimum error can be found using gradient descent. These points of minimal error are then fed back into the weights of the network

and network runs again. This repeats until the minima for the error is discovered. This process of improving the accuracy of weighted values over time is known as backpropagation. Finally, one of the components of the neural network known as a activation function discards values that are not with a defined range. The end result is a higher-dimensional curve that can be used to classify data based on its output from the algorithm. The goal of DNNs is to mimic the biological concept of neurons and synaptic firing.

Methodology The data for this research was collected using a Python script for Blender that gets the global bone positions of each and writes them to a CSV file. The poses are also labeled in the script. The CSV file is then manually converted to an Excel file. In the case of the DNN, the algorithm is implemented using TensorFlow and Keras with Python which organized the data and then apply predefined layers, respectively. In the case of the SVM, distance between the line was optimized using Lagrangian and was tested on each class independently vs the neutral pose. In the case of the Naïve Bayes, the associated algorithm was a standard implementation. Both the SVM and Naïve Bayes are implemented using R. Two main iterations of the data were attempted. The first iteration included the bone positions in the dataset as well as the transformations, while the second only included the Euclidean rotations of each bone. In both cases, the data also included the name of the bone in question as well as label for the action that bone was participating in. This was done to account for the central hypothesis of this research.

Data

Analysis

Conclusion and Future Research As can be concluded from the methods attempted, the lowest level machine learning algorithm that can handle the task of pose classification is a DNN. The two approaches have some inherent issues with handling the necessary animation data. The SVM loses the advantage of efficiency because it is limited to a two-dimensional separation of the classes. The Naïve Bayes algorithm also failed to handle the task adequately because the differences between poses are both small and important. The accuracy of the algorithms peaked at 80%, which, given four classes, is 320% better than random guessing. In future research, more complicated algorithms could be utilized to potentially achieve better results. Research could also be conducted on treating actions a sequential rather than discrete data could also be conducted. Certain actions, such as handwaving can only be classifying in motion. Motion implies sequence. However, such research was beyond the scope of this work. The decimal accuracy of bone positions is a persistent issue but can be handled by considering the bones transformations exclusively. Therefore, the null hypothesis that bones transformations are not the main source of pose difference can be rejected.

References