SOFTWARE REQUIREMENTS SPECIFICATION

for

CS 4ZP6 Capstone Project

Version 0.0

Prepared by Brendan Duke, Andrew Kohnen, Udip Patel, David Pitkanen, Jordan Viveiros

McMaster Text to Motion Database

October 10, 2016

Contents

Revision History

Name	Date	Reason For Changes	Version
Brendan Duke	Oct. 7th, 2016	Initial Version	0.0

1 Project Drivers

1.1 The Purpose of the Project

1.1.1 The User Business or Background of the Project Effort

With the current advancement of deep learning architectures, specifically the use of Recurrent Neural Networks used for sequential data processing like natural language or pose estimates. Using a Recurrent Neural Networks is made more available through the creation of large databases that contain video with descriptive labeling and annotations like MovieQA, Charades, or MSR-VTT. Expanding on these technologies can be utilized to build a system for "Computational Storytelling" that takes in a short story of 5 lines and outputs an animated video between an AI and human director.

1.1.2 Goals of the Project

The goal of this project is to support a text to motion subcomponent of a larger collaboration between the University of Guelph, SRI, and other institutions. The creation of a database, website and manipulation of already established pose estimation software will be required. Creating this database and website will allow the larger text to motion project use the relationship developed through the pose estimation in order to provide a pose and word pairing for animating.

1.2 The Client, the Customer, and Other Stakeholders

The current stakeholders in this project are the: - Supervisors of the project (Dr. He, Dr. Taylor) - University of Guelph "text to motion" research group - The McMaster Capstone group

1.2.1 The Client

The current clients for this project are Dr. Taylor and his graduate student Thor Jonsson. Dr. Taylor is the primary driver to develop a website and database where annotated motion information could be generated and pulled from as a growth point into the larger text to motion project. They will be using the database to create Recurrent Neural Networks that will pair actions and their pose found within the database to words or combinations found in the input story.

1.2.2 The Customer

The customers are included within the clients since building this database and website combination will be utilized by Dr. Taylors research team and their external partners. In addition to Dr. Taylor and his research team this project would appeal to anyone that needed a pairing of actions and pose estimations as the website would be readily available to others.

1.2.3 Other Stakeholders

In addition to the stakeholders listed above the success of the project could bring in additional stakeholders that are interested in the method of pose estimation, database information or want to use this information for a project of their own. This could be another research group or deep learning framework that is looking to expand.

1.3 Users of the Product

1.3.1 The Hands-on Users of the Product

The primary users of this project will be Dr. Taylor and his research team as they need this data in order to complete the Computational Storytelling. They are also comfortable with the process of deep learning and would provide input on how the pairings should be set up.

1.3.2 Priorities Assigned to Users

1.3.3 User Participation

1.3.4 Maintenance Users and Service Technicians

2 Project Constraints

2.1 Mandated Constraints

2.1.1 Solution Constraints

Constraint Number	0
Constraint Type	4a. Solution Constraint
Event/Use Case Numbers	Entire product.
Description	The Text-to-Motion Software Suite must run under Linux.
Rationale	Linux is the operating system used by the Guelph Machine Learning research lab, and also the most commonly used operating system in the research community.
Originator	Dr. Graham Taylor
Fit Criterion	Automated builds and testing should pass on popular Linux distributions: Ubuntu, Fedora and RHEL.
Customer Satisfaction	5
Customer Dissatisfaction	5
Priority	High priority.
Conflicts	None.
Supporting Materials	None.
History	Created September 26th, 2016.

Constraint Number	1
Constraint Type	4a. Solution Constraint
Event/Use Case Numbers	Entire product.
Description	Major APIs to the Text-to-Motion database must be accessible from the Python programming language.
Rationale	Python is a popular, easy-to-use, and quick-to-prototype language, and is therefore one of the most favoured programming languages among the Machine Learning research community.
Originator	Dr. Graham Taylor
Fit Criterion	There must be hooks to all major interfaces written in Python, and there must be tests that are directly testing the Python interfaces.
Customer Satisfaction	5
Customer Dissatisfaction	5
Priority	High priority.
Conflicts	None.
Supporting Materials	None.
History	Created September 26th, 2016.

Constraint Number	2
Constraint Type	4a. Solution Constraint
Event/Use Case Numbers	Human Pose Estimation Event.
Description	The human pose estimation component should use deep learning methods.
Rationale	This constraint is to allow Dr. Taylor's group to integrate the software into their existing text-to-motion pipeline
Originator	Dr. Graham Taylor
Fit Criterion	Dr. Taylor should confirm that the deep learning methods used in the human pose estimator are satisfactory.
Customer Satisfaction	5
Customer Dissatisfaction	4
Priority	High priority.
Conflicts	None.
Supporting Materials	None.
History	Created September 26th, 2016.

Constraint Number	3
Constraint Type	4a. Solution Constraint
Event/Use Case Numbers	Entire product.
Description	The project must be completed by April 5th, 2017.
Rationale	The project is part of the CS 4ZP6 Capstone Project course.
Originator	Dr. He
Fit Criterion	All documentation, testing and implementatoin must be completed and checked in to GitHub by April 5th, 2017.
Customer Satisfaction	5
Customer Dissatisfaction	5
Priority	High priority.
Conflicts	None.
Supporting Materials	None.
History	Created September 21st, 2016.

2.1.2 Implementation Environment of the Current System

2.1.3 Partner or Collaborative Applications

2.1.4 Off-the-Shelf Software

2.1.5 Anticipated Worklace Environment

2.1.6 Schedule Constraints

2.1.7 Budget Constraints

2.2 Naming Conventions and Definitions

2.2.1 Definitions of All Terms, Including Acronyms, Used in the Project

The Project when used, is referring to the McMaster Text to Motion Database project. The project aims to generate a database of human pose estimation model information that is linked to videos of human motion containing rich text annotations.

Human Pose Estimation is the process of estimating the configuration, or pose, of the body based on a single still image or a sequence of images that comprise a video. Human pose estimation may find the chin, radius, humerus, and other bone and joint positions.

Feedforward Neural Networks are artifical neural networks where connections between the units do *not* form a cycle). They are the simplest type of neural network, because information moves in only one direction.

ConvNets or **Convolutional Neural Networks** are a type of feed-forward artificial neural network. ConvNets are inspired by the visual cortex and are commonly used in visual recognition applications.

RNNs or **Recurrent Neural Networks** are a class of artificial neural networks where units form a directed cycle, in contrast with feed-forward neural networks.

Deep Belief Networks are a type of deep neural network composed of multiple layers of "hidden units" (variables that are not observable), with connections between layers but not between units of a given layer.

2.2.2 Data Dictionary for any Included Models

2.3 Relevant Facts and Assumptions

2.3.1 Facts

2.3.2 Assumptions

3 Functional Requirements

3.1 The Scope of the Work

3.1.1 The Current Situation

There is a large amount of existing research into human pose estimation, which this project will leverage. Based on constraint 2, we focus on existing solutions that use deep learning methods.

[1] present a ConvNet architecture for human pose estimation from videos, which is able to benefit from temporal context across multiple frames using optical flow. This work is focused on upper-body human pose estimation only.

[2] propose a ConvNet model for predicting 2D human body poses in an image. This model is able to achieve state-of-the-art results using a simple architecture, and draws on the work done in [1].

[3] introduces *Convolutional Pose Machines (CPMs)* for pose estimation in images. CPMs consist of a sequence of ConvNets that iteratively produce 2D belief maps.

3.1.2 The Context of the Work

3.1.3 Work Partitioning

Table 3.1: Business Event List

Event Name	Input and Output	Summary
Web Interface Skeleton Overlay	**IN**: An image or video with humans in it. **OUT**: The same image or video, with a skeleton overlaid on top of all humans indicating their bone and joint positions.	Allow users to observe the human pose estimation component in real time through a web interface.
Web Interface Text-to-Motion	**IN**: Word or phrase describing a human pose or action. **OUT**: Rich-text-annotated video corresponding to the input word/phrase, complete with overlaid skeleton.	Allow users to see the output of searches on the database using pose and/or action keywords, such as "run" or "kneeling".
Database Interface Skeleton Overlay	**IN**: A stream of video with humans depicted. **OUT**: A set of human pose estimations corresponding to the video, in a standard data format.	Users should be able to use the human pose estimation solution to generate their own motion data set.
Database Interface Text-to-Motion	**IN**: Word or phrase describing a human pose or action. **OUT**: Video in common encoding (e.g. MP4), associated rich-text-annotations, and human pose estimations in a standardized format.	Provide users direct access to the raw motion-estimation data format based on action-keyword database lookup.

3.2 The Scope of the Product

3.2.1 Product Boundary

3.2.2 Product Use-case List

3.2.3 Individual Product Use Cases

3.3 Functional and Data Requirements

3.3.1 Functional Requirements

Requirement Number	4
Requirement Type	9a. Functional Requirement
Event/Use Case Numbers	
Description	The text-to-motion software suite will provide an API to read individual frames in RGB format from a video stream. At least MP4, MP2 and AAC must be supported.
Rationale	Researchers may wish to do their own processing on RGB frames before feeding those frames into the human pose estimation module.
Originator	Brendan Duke.
Fit Criterion	For a given set of test video streams, the frame-capture API must produce RGB frames identical to known reference frames.
Customer Satisfaction	3
Customer Dissatisfaction	3
Priority	Moderate priority.
Conflicts	None.
Supporting Materials	None.
History	Created October 5th, 2016.

3.3.2 Data Requirements

4 Nonfunctional Requirements

4.1 Look and Feel Requirements

4.1.1 Appearance Requirements

4.1.2 Style Requirements

4.2 Usability and Humanity Requirements

4.2.1 Ease of Use Requirements

4.2.2 Personalization and Internationalization Requirements

4.2.3 Learning Requirements

4.2.4 Understandability and Politeness Requirements

4.2.5 Accessibility Requirements

4.3 Performance Requirements

4.3.1 Speed and Latency Requirements

4.3.2 Safety-Critical Requirements

4.3.3 Precision or Accuracy Requirements

4.3.4 Reliability and Availability Requirements

4.3.5 Robustness or Fault-Tolerance Requirements

4.3.6 Capacity Requirements

4.3.7 Scaling of Extensibility Requirements

4.3.8 Longevity Requirements

4.4 Operational and Environmental Requirements

4.4.1 Expected Physical Environment

4.4.2 Requirements for Interfacing with Adjacent Systems

4.4.3 Productization Requirements

4.4.4 Release Requirements

4.5 Maintainability and Support Requirements

4.5.1 Maintenance Requirements

4.5.2 Supportability Requirements

4.5.3 Adaptability Requirements

4.6 Security Requirements

4.6.1 Access Requirements

4.6.2 Integrity Requirements

4.6.3 Privacy Requirements

4.6.4 Audit Requirements

5 Project Issues

5.1 Open Issues

5.2 Off-the-Shelf Solutions

5.2.1 Ready-Made Products

5.2.2 Reusable Components

5.2.3 Products That Can Be Copied

5.3 New Problems

5.3.1 Effects on the Current Environment

5.3.2 Effects on the Installed Systems

5.3.3 Potential User Problems

5.3.4 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

5.3.5 Follow-Up Problems

5.4 Tasks

5.4.1 Project Planning

5.4.2 Planning of the Development Phases

5.5 Migration to the New Product

5.5.1 Requirements for Migration of the New Product

5.5.2 Data That Has to Be Modified or Translated for the New System

5.6 Risks

5.7 Costs

5.8 User Documentation and Training

5.8.1 User Documentation Requirements

5.8.2 Training Requirements

5.9 Waiting Room

5.10 Ideas for Solutions

Bibliography

[1] T. Pfister, J. Charles, and A. Zisserman, "Flowing convnets for human pose estimation in videos," *CoRR*, vol. abs/1506.02897, 2015. [Online]. Available: http://arxiv.org/abs/1506.02897

[2] V. Belagiannis and A. Zisserman, "Recurrent human pose estimation," *CoRR*, vol. abs/1605.02914, 2016. [Online]. Available: http://arxiv.org/abs/1605.02914

[3] S. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," *CoRR*, vol. abs/1602.00134, 2016. [Online]. Available: http://arxiv.org/abs/1602.00134