

## AUTONOMOUS ANIMATION IN EMBODIED AGENTS

### TECHNICAL FIELD

- [1] Embodiments of the invention relate to autonomous animation of Embodied Agents, such as virtual characters, digital entities, and/or robots. More particularly but not exclusively, embodiments of the invention relate to the automatic and real-time analysis of conversational content to dynamically animate Embodied Agents.

### BACKGROUND ART

- [2] Behaviour Mark-up Language, or BML, is an XML-based description language for controlling verbal and nonverbal behaviour for “Embodied Conversational Agents”. Rule-based gesture generators, such as BEAT (SIGGRAPH '01) apply rules to generate gestures, paired with features of text, such as key words. This results in repetitive and robotic gesturing, which is difficult to customize on a granular level. Large databases of rules and gestures are required. Speech-driven gesture generators use neural networks to generate automatic movements from learnt gesture and speech combinations. However, these generators often work in a black-box manner, assume a general relationship between input speech and output motion, and have been of limited success.
- [3] US9205557B2 discloses a method for generating contextual behaviours of a mobile robot. A module for automatically inserting command tags in front of key words is provided. US9721373B2 discloses programs for creating a set of behaviours for lip sync movements and nonverbal communication which may include analysing a character's speaking behaviour with acoustic, syntactic, semantic, pragmatic, and rhetorical analyses of the utterance.
- [4] Efficient, automatic on-the-fly augmentation and/or modification of communicative utterances by embodied, autonomous agents remains an unsolved problem. Further, animating Embodied Agents in a manner that is realistic, non-repetitive and readily customizable remains an unsolved problem.

### OBJECT OF INVENTION

- [5] It is an object of the invention to improve autonomous animation in embodied agents, or to at least provide the public or industry with a useful choice.

## BRIEF DESCRIPTION OF DRAWINGS

Figure 1 shows a TTG Algorithm according to one embodiment.

Figure 2 shows a Emphasis Detection algorithm according to one embodiment.

Figure 3 shows an example of suitable rules weightings for Emphasis Detection.

Figure 4 shows an example of the scoring process for Emphasis Detection.

Figure 5 shows an Embodied Agent in a variety of different Poses.

Figure 6 shows blending between arm Variation Poses.

Figure 7 shows a first example of blending between hand Variation Poses.

Figure 8 shows a second example of blending between hand Variation Poses.

## DISCLOSURE OF INVENTION

- [6] Embodied Agents, such as virtual characters, digital entities, and/or robots may interact with a user by uttering speech from textual input in real-time. An Embodied Agent may be a digital avatar, cartoon character, anthropomorphic avatar, etc., or may be a physical avatar e.g., physical robot, etc. A physical robot may include various mechanical units for different parts, e.g., face part, body part, etc, enabling the physical avatar to make various facial motions and/or body motions.
- [7] An Embodied Agent may have a face comprising at least one of eyes, nose, mouth, and may be animated to present various facial motions. The avatar may also have one or more body parts, including at least one of a head, shoulders, hands, arms, legs, feet, etc., and may be animated to present various body motions.
- [8] Text to speech (TTS) and lip animations synchronized to the speech enable such Embodied Agents to resemble human-like speech. Nonverbal communication, such as facial expressions and hand-gestures assist with human communication and bring realism and to the animation of Embodied Agents.
- [9] Embodiments described herein relate to the autonomous animation of Gestures by the automatic application of animations to Input Text – or the automatic application of animation Mark-up wherein the Mark-up triggers nonverbal communication expressions or Gestures.

### **Text-To-Gesture**

- [10] In order for an Embodied Agent's movements to come across as natural and human-like as possible, a Text-To-Gesture Algorithm (TTG Algorithm) analyses Input Text of a Communicative Utterance before

it is uttered by a Embodied Agent, and marks it up with appropriate and meaningful Gestures given the meaning, context, and emotional content of Input Text and the gesturing style or personality of the Embodied Agent.

- [11] For example: The Input Text: “*Would you like to talk about our technology, or our business model?*” may be processed by the TTG Algorithm to output -> “*#SlightlyHappy Would you #Shrug like to #Smile talk about our #BeatBothArmsLeft technology, or our #BeatBothArmsRight business #PalmsSpread model?*”
- [12] The TTG Algorithm uses Natural Language Processing (NLP) to get the best possible understanding of the context, meaning and communicative intention from the about-to-be-uttered-text, in order to generate the most nuanced and natural gestures for it. The TTG Algorithm is modular and extensible, so that new and more sophisticated analysis can be added, and existing analysis can be modified or removed easily.

### **Method**

- [13] Figure 1 shows a TTG Algorithm according to one embodiment.

#### *Parsing*

- [14] At step 102, Input Text 6 is received by a Parser which returns a Parse Tree for each clause of the Input Text 19. Each clause is a tree, and each node in the tree is a token, roughly equivalent to a word, and also contains information about the token such as its lemma, part of speech tag, and the dependency relationship with its parent node, whether it is a strong keyword, part of a list of noun phrases, etc. In one embodiment, dependency parsing outputs a dependency tree, which provides relationships between tokens. Any suitable dependency parsing method or system may be used.

#### *Clause Analyser*

- [15] At step 104, a clause analyser attaches further information about the Input Text 19 to the Parse Tree 8. The clause analyser derives information about the clause and tokens, to provide as input to the Mark-up Generator which generates Mark-up based on clause analysis information.
- [16] Clauses are analysed for semantic and syntactic patterns, keywords, emotions and dialogue acts are identified. In one embodiment, the clause analyser receives a dependency tree and using the dependency information identifies beats, negations and enumeration behaviours in the clause. Clause Analysis also attaches sentiment information to the dependency tree.

#### Sentiment

- [17] Any suitable machine learning or rule-based method may be used to classify the sentiment of the clause. Clauses may be classified based on valence (positive-neutral-negative), arousal (low-neutral-high), and fine-grained emotional content (for example: joy, sadness, anger, surprise, fear, disgust).

- [18] In one embodiment, a text sentiment analysis function is constructed using a support vector machine (SVM). Any suitable method of text sentiment analysis may be used. The SVM may be trained using conversational content from a specific domain. For general purpose conversation, the SVM may be trained using a broad range of domains and styles, lengths of utterance and other parameters. Any other suitable classifier may be used, including, but not limited to, a neural network, decision tree, a regression-based classifier, a Bayesian classifier. A deep neural network may be suitable for classifying fine-grained emotional content.
- [19] Word Sentiment may identify sentiment at the word level and identify words as positive or negative. In one embodiment, a negative/positive word dictionary is used. The valence of individual words in a clause may be recorded. For example, in a clause with an overall positive valence, the clause analyser may identify on-negated words with positive valence words, and non-negated words with a negative valence.
- [20] In one embodiment, sentiment-based animations are applied to sentences, based on the sentiment score. Any suitable model for sentiment analysis may be used and appropriately trained to determine a sentiment score.

#### Negation Scope Detection

- [21] Tokens (words) that are negated can be determined based on dependency links (e.g., descendants of a negation are considered to be negated by the negation). The dependency tree structure may determine the scope of any negation words (i.e. which words can be considered negated). In particular, any word that is a descendant, a sibling, or a nibling (a child of a sibling) of a negation falls within the scope of the negation.

#### Enumeration

- [22] Noun chunks and phrasal verbs may be used to determine groups of words. A list of noun chunks (noun phrases) may be provided.
- [23] Phrasal verbs may be detected. In one embodiment, phrasal verbs may be detected by an algorithm comprising the steps of: 1. Finding verbs, 2. Searching backwards for adverbs, 3. Searching forwards for adverbs and preposition and noun phrases.
- [24] Information about groups of words may be used to drive animation. For example, in “Would you like a green avocado, or a brown avocado?” an embodied agent could point to the left over “green avocado” and right over “brown avocado”, rather than treating them as individual words.

- [25] Beats may be repeated within a group. For example: "I am going on holiday tomorrow" might trigger a circle on 'going' and a chop on 'tomorrow', but "I can see a big yellow fluffy giraffe" might trigger repeated chops on 'big' 'fluffy' and 'giraffe'.

#### Dialogue act classification

- [26] Dialogue act classification may classify dialogue acts such as listing options, asking a question, explaining, offering alternatives, describing, asserting, retracting, offering an opinion, apologizing, greeting, changing the subject, predicting, instructing, explaining, insulting, or teasing. In other words, dialogue act classification classifies what a Communicative Utterance is trying to achieve.
- [27] Dialogue act classification may be carried out using any suitable classification method, including, but not limited to, rule-based methods and machine learning based methods. In one embodiment, a deep learning classifier is trained on a broad range of dialogue acts.
- [28] For questions, the grammatical mood of the Communicative Utterance may be determined (questions tend to be in the interrogative mood), or check it against a dictionary of 'question' phrases, like beginning with who, what, when, where, how, do, does. The dialogue act classifier may also receive as input whether there is a question mark at the end of the clause. The dialogue act classifier may subdivide this dialogue act into different kinds of questions, like asking the user about themselves or for their opinion, asking for clarification, asking to repeat, and rhetorical questions. Advising and instructing are often in the imperative mood, or preceded by "you should" or "you could".
- [29] For offering alternatives or contrasting ideas, it might be two clauses separated by a conjunction such as 'or' or 'but', or two noun phrases or phrasal verbs separated by a conjunction. For example, "*We could organise a party for him, or we could wait and see if he organises one himself*". For listing several options or items, find a series of noun phrases or phrasal verbs separated by commas or conjunctions. For example, "Are you going on holiday or travelling for work?"; "You will need a pair of 3mm needles, 100g of 4ply yarn in the colourway of your choice, and a cable needle."
- [30] In another example, if the text is "there are many banks in New Zealand: ASB, ANZ, BNZ and Westpac.", the intention may be classified as "enumeration". Hypotheticals, conditionals, or counterfactuals may be indicated with the phrases "what if", "only if", "if... then..." and so on.
- [31] In one embodiment, dialogue act classification can be combined with sentiment analysis to add further nuances to nonverbal communication.

#### Tone Classification

- [32] The tone of the content being delivered may be classified and used to modulate behavioural performance. Examples of dimensions of tone to classify may include serious vs. humorous, deferent vs. assertive, aloof vs. compassionate, casual vs. formal, or matter of fact vs. enthusiastic. In other words, tone classification classifies the manner of a Communicative Utterance and may modulate gestural and emotional performance while the utterance is delivered accordingly.
- [33] Tone classification may be carried out using any suitable classification, including, but not limited to, rule-based methods and machine-learning based methods. In one embodiment, different dimensions of tone may be classified via different machine learning classifiers. In another, a deep learning classifier may classify across a wide range of tonal dimensions.

#### Pattern Analysis

- [34] A pattern analyser finds noun phrases and series of noun phrases, phrasal verbs and series of phrasal verbs. The pattern analyser may identify transitive verbs by checking for a preposition and then a noun phrase, following the rest of the verb phrase. For example, “very quickly running away from the wolf” is analysed as a phrasal verb, because the part of speech tags are, respectively, “ADVERB, ADVERB, VERB, ADVERB, PREPOSITION, DETERMINER, NOUN” (and ‘DETERMINER, NOUN’ is a noun phrase).
- [35] The pattern analyser may determine the mood, tense, verb form, adjectival form (eg superlative, comparative), person, number and other morphological features.
- [36] Such information may be used to influence animation – for example, by increasing the size of gestures on superlative and comparative tokens.
- [37] The “person” of a clause may influence animation by animating actions such that they are directed to the appropriate “person”. For example, a clause in first person may generate more speaker-directed actions, a clause in second person may generate more listener-directed actions and a clause in third person may generate undirected actions.
- [38] Tense of a clause may influence gestures, for example, by animating clauses in the past tense with more “rigid” animations, and the future tense with “looser” animations, representing hypotheticals.

#### Location Analysis

- [39] Dictionaries of positional and directional phrases may be provided: one for each of high, low, narrow (or centre) and wide. These can be exact string matches or pattern matches, for example “under \$NOUNPHRASE” would match to “he was under the sea”, “it was under a big round table” and “she was there under some kind of pretence” but not “they were under 18”.

### Context

- [40] Contextual information from previous clauses and even previous conversation turns (both the Embodied Agent 12 and the user's) to provide broader context for the particular clause being analysed. For example, if the Embodied Agent 12 is asking the user to repeat themselves, the Embodied Agent 12 may perform the utterance slightly differently the second time around: with more emphasis on the key points or with less hesitancy. If a proper noun or other term has already been introduced into the conversational context, it may be less likely to be a keyword in subsequent mentions. If the current utterance is changing the topic of conversation, there may be more (or larger or more forceful) gestures to indicate emphasis on the new topic.

### *Mark-up Generator*

- [41] At step 108, a Mark-up Generator uses the information in the Analysed Tree to generate Mark-up for various kinds of gestures. The Analysed Tree may comprise a Parse Tree annotated with information from Clause Analysis. Each of these Mark-up generators may add candidate Mark-ups before or after the whole clause, or before or after any individual word. Many gestures 'hit on' (or 'are triggered on') a word, by which we mean that the stroke point of the gesture (extreme point) occurs at the same time as the stressed syllable of that word. This means that the gesture may start before the word, in order to give it time to reach its stroke point at the moment of the stressed syllable.
- [42] Gestures include facial expressions, head and neck gestures, arm and hand gestures, and full body movement. All gestures are made up of a pose and an action, where the pose is the starting point of the gesture and the action is the motion applied from that starting pose. For each action, the starting pose may be defined explicitly or it may be whatever the current pose is, for example the end pose of the previous gesture.

### Dialogue act specific gestures

- [43] Examples of dialogue act specific gestures which may be applied include: Questions triggering shrugs and palms up outward arcs, triggering on the main verb or keyword of the clause. Negations trigger head shakes and arms crossing over or wrists flicking dismissively. Offering alternatives map to one hand out to one side and then the other to the other, indicating weighing scales. Listing nouns or verbs as, for example, three options, map to both arms gesturing with a chop to one side, then both in the middle, then both to the other side (or similar gestures that follow a path, such as pointing at a low level, then a bit higher, then a bit higher still). Any more than four items in a list instead maps to counting off on the fingers.

### Symbolic Gestures

- [44] Symbolic gestures are those that carry a specific meaning. The meaning might be emblematic (the gesture stands in for a word or phrase), for example a wave communicating a greeting; iconic (the gesture literally represents the meaning of the word or phrase), for example tracing a square shape for the word “box”; or metaphoric (the gesture represents the meaning of the word or phrase, but not literally), for example tracing a square shape for the the word “confined”. These are triggered from a dictionary lookup for each gesture, containing emblematic, iconic and metaphoric trigger phrases in one dictionary. The phrases in the dictionary can optionally indicate which word in the phrase the gesture should hit on. By default, it will hit on the first word in the phrase. These phrases can also be matched to patterns rather than exact string matches, for example “I am Sam” matches the pattern “I am \$PROPERNOUN”, but “I am hungry” does not. This kind of gesture should be applied sparingly, otherwise it can look like they are acting out the utterance, which can come across as comical or patronising. The rate of symbolic gestures is defined in the personality/style configuration. In one embodiment, symbolic gestures match against a universal dictionary for each gesture.

### Beats

- [45] Beat gestures emphasise words non-meaningfully (e.g. not in a symbolic way or connected to any specific dialogue act). Beats are triggered on words in the clause as picked out by the Emphasis Detection algorithm, at a rate defined in Configuration Settings. The Action is chosen based on the personality and gesturing style as defined in the config. The kinds of Actions include chops (up, down, diagonal), circles, and arcing actions, all of which can be applied on a range of base arm and hand Poses to produce a wide variety of gestures: from a rigid pontificating gesture to a fluid open arcing gesture.
- [46] Thus Beats are applied to keywords as specified in the analysed tree, of types defined in global Configuration Settings. Each beat gesture consists of a pose and an action, and each pose consists of arm, wrist, and hand elements.

### Embodiment Gestures

- [47] Embodiment Gestures are gestures that people do in virtue of being embodied. For example, people take a deep breath or sigh before starting a long description or explanation. In Embodied Agents, deep breaths may be triggered before long sentences. Another example is shifting weight from one foot to the other, which occurs when people get tired. In Embodied Agents, this may be triggered between (some) clauses and in other gaps. Pausing and looking up to one side to think or remember something, may be triggered stochastically between clauses and before long or very rare words, or proper nouns the first time they are used, as if trying to think of the word or name. Sometimes these are accompanied by a furrowed brow, or a filled pause or hesitation marker such as ‘um’. People do a wide array of grooming gestures, such as

straightening their clothes, scratching their noses, or tucking their hair behind their ears, which are triggered in gaps with no other gestures, at a rate that is specified by the personality of the individual.

#### Turn-Taking Gestures

- [48] When people pause their speech but don't intend to cede the conversational floor, they tend to look away and sometimes do floor-retaining gestures (such as holding up a hand or a finger), or fill the pause with an 'um' or 'ah'. Turn-taking behaviour may be triggered at some clause boundaries and before long or rare words, including proper nouns the first time they are mentioned. When people have finished speaking, to cede the floor, they (for example) make direct eye contact and smile expectantly, sometimes also doing a 'your turn' type gesture (for example, one or both hands with palms up indicating towards the conversational partner). Such gestures are triggered at the end of their entire utterance (which may be one or several clauses). When a conversational partner (user) attempts to interrupt the character, they might do a floor-retaining gesture to indicate they're not giving up the floor, or they might look a bit surprised and stop talking and gesturing, ceding the floor to the user (how likely they are to do this may be configurable based on personality and role). When a user is speaking, backchannel gestures are triggered in the form of nods and smiles, frowns, 'hmm's and 'uh huh's, based on rapid sentiment analysis of interim STT results.

#### Poses

- [49] A pose is the starting point for a gesture: where the body moves to before it starts the gesture. For example, poses may include body, head, arm, wrist and finger elements. Each of these may have a base pose and some controlled random variation added on. Each element is chosen from a set of base poses that are compatible with the chosen action (as the action is the main part of the gesture, it is chosen first). From these compatible poses, the pose is chosen stochastically at frequencies defined by the personality, style and role config. Controlled random variation is obtained by blending in a small amount of a "variation pose". These variation poses are chosen using information from the location analyser, as well as the sentiment scores, and if not determined by those, is chosen at random. The amount of the variation pose that is blended is chosen from a range specified either by the location analyser, sentiment modulation, or the default range (which is likely to be smaller values since it's just for adding variety, not visibly pulling the pose in a specific direction).

#### Voice Modulation

- [50] Tags may be inserted to modulate the voice in order to align it better with the gestures chosen; the result being a coherent overall performance of the utterance. For example, speed, pitch, and volume of the voice on individual words may be modified to emphasise those words. Such features may be modulated for an

entire clause to change the emotional tone. For example, increasing speed and volume while decreasing pitch sounds more angry, decreasing all three makes it sound more sad, etc.

### *Mark-up Solver*

- [51] The Mark-up solver takes the Parse Tree which has been annotated with all the candidate Mark-ups as decided by the respective Mark-up generators, and outputs the original text with appropriate Mark-ups added to produce a coherent performance of that utterance to be sent to be processed into speech and animation.
- [52] For example, some gestures can be performed together (like one head gesture and one body gesture), while others cannot. Some gestures only make sense to perform in conjunction with a series of other gestures (for example, if the utterance was “on one hand, A, but on the other hand, B” it makes the most sense to do both sides of the gestures for weighing up two options, rather than doing one side and not the other). This Mark-up solver resolves these conflicts but retains connected gestures, to build a coherent gestural performance of the utterance.
- [53] In one embodiment, for words that have at least one Mark-up tag, the Mark-up Solver picks at most one body gesture and one head gesture for each word. This may be implemented using a priority-based approach. Where there are multiple candidate gestures for a given word, gestures may be chosen in a predefined order of priority. In one embodiment, the following order of priority is used:
- Replacing existing manual tags
  - Client-override tags
  - Symbolics, but not too many
  - Dialogue acts
  - Enumerating
  - Beats
  - Turn-taking
  - Embodiment
- [54] In another embodiment, the whole clause or even the whole paragraph is taken into account, to ensure that the gestures taken as a whole formed a coherent performance. It would ensure that a series of gestures taken together formed a sequence in a sensible or natural pattern. For example, a wide arcing gesture followed by one or more small chop beats is a common sequence, but a chop then an arc then another chop is less natural, and a series of gestures that zigzag in space (wide, narrow, wide, narrow) tends to look unnatural unless they are zigzagging for a communicative (symbolic) reason. It would also ensure

that longer or more significant gestures were given enough time to play out but shorter gestures could be triggered in faster succession.

### **Pose and Action Scheme**

- [55] A Pose and Action scheme independently applies Pose and Action to Input Text. Independent Poses and Actions may be applied to beat gestures, or any other suitable type of Gesture.

#### Pose

- [56] A Pose is the dimensions and/or position of a Gesture, such as the position of limbs. For example, the Pose of the arms of a Embodied Agent (e.g. arm positions) may be wide/narrow, or high/low.

- Wide/Medium/Narrow
- High/Medium/Low

- [57] Figure 5 shows an Embodied Agent 12 in a variety of different Poses while the Embodied Agent 12 speaks a Communicative Utterance. The Input Text and Mark-up of the Communicative Utterance is as follows:

[middle\_pose][strong\_beats] Please place your [low\_beats] ticket [low\_pose] under the [medium\_beats] scanner.

- [58] The example shows how Poses and Actions may be applied at different parts of Input Text. Once a pose is defined, all subsequent actions start from the defined pose.

- [59] Figure 5A shows the Embodied Agent 12 in a wide arm / medium arm height Pose. Figure 5B and Figure 5C show the Embodied Agent 12 in a low arm height Pose.

Poses may be associated with a pose speed (how quickly a certain pose is reached from a neutral pose or previous pose). Poses may be associated with property tags, e.g.:

- String name
- Left/Right or both (referring to whether the pose is a handed one)
- Dimension tags. For example, Arm poses may be associated with a width tag (e.g. whether it is a narrow, medium, or wide width pose) and/or a height tag (whether it is a high, medium, low height arm pose).

- [60] In one embodiment, the Embodied Agent 12 is returned to a “neutral” pose after each action. In another embodiment, the end pose of a certain action may become the new start pose of a new Action.

### Actions

- [61] Actions refer to the movement trajectories of various feature points in a face part or a body part. Actions may be based on any suitable 3D reconstruction techniques. For example, an Action indicating a body motion may be reconstructed by a set of predetermined feature points in the body part.
- [62] Actions may be configured with suitable parameters, including, but not limited to:
- Type
  - Intensity
  - Frequency
  - Speed
- [63] one property of each action is which poses it can be applied on top of (not all combinations work, for example if the pose is already wide and the gesture is opening the arms out wide).

### **Word-Token-Matching Regular Expression**

- [64] A regular expression is a sequence of characters that specify a search pattern. These patterns can be used by text-searching algorithms to find instances of text that match the pattern. Modern regular expressions used in computing are called 'regex' and typically include (but not limited to) these operators:
- Normal text characters and numbers: a-z, A-Z, 0-9, CJK characters, spaces etc. e.g. a search pattern "a" will match the text "cat" at the second position. E.g. A search pattern "cat" will match the text "concatenate" at position 4.
  - '.': A dot is a wildcard. It will match any character. E.g. A search pattern "c.t" will match "cat", "cot", and "cut".
  - '\*': An asterisk will match zero-or more of the preceding character. E.g. A search pattern "cut\*" will match zero or more of the 't' character: "cube", "cute", "cutting"
  - '+': A plus sign will match one-or-more of the preceding character.
  - '()': Parentheses define scope and precedence of operators.
- [65] In one embodiment, a method of text-matching operates on clause tokens instead of individual characters.
- [66] A "token" normally corresponds to an individual word with some exceptions: "don't" resolves to two tokens representing "do" and "n't". Grammatical particles, such as a comma ",", have dedicated tokens. These tokens encapsulate linguistic features of the text they represent as attributes, including (but not limited to):

- Part-of-speech: Noun, Verb, Adjective, Punctuation etc. These can be specified as a standard shorthand: "adjective" is "ADJ", "proper noun" is "PROP\_N" etc.
- Detailed part-of-speech: comparative adverb, determiner, proper singular noun, etc.
- Lemma: The base form of the word. E.g. the lemma of "looking" is look. The lemma of "is" is "be".
- Stem: The word stem (Not currently used in any form. Could be used in the future). E.g. The stem of "fishing", "fished", and "fisher" is "fish". The stem of "argue", "argued", "argues", and "arguing" is "argu".
- Dependency: The syntactic dependency, or the relationship of a token to its parent token (Tokens exist within a tree structure and each token may have a parent or children).

[67] Ordinary text can be converted into tokens using any suitable tool, such as SpaCy.

[68] This token-based text matching can be used by specifying an attribute to match with. For example:

- "\$lemma:look over there" will match "They looked over there", "They are looking over there", and "They will look over there".
- "I am \$pos:PROP\_N" will match proper nouns, e.g. the character introducing themselves: "I am Sam", "I am Rachel" etc.
- "was \$pos:ADV+ excited" the '+' symbol will match one-or-more of the preceding operator (adverb). e.g. "I was really excited", "I was really very excited" - "really" and "very" are both adverbs in this sentence.
- The asterisk operator can be used similarly to match zero-or-more: "was \$pos:ADV\* excited" will additionally match "I was excited".
- "a . or ." the '.' symbol here will match any token, unlike in normal regular expressions where it would match a single letter/numeral. "a . or ." could be useful for detecting when alternatives are being presented.

[69] Dictionary files storing lists of these search patterns may be stored. If some text matches one of the search patterns, a relevant action or emotion may be registered to be performed when that text is spoken.

### **Configurability**

[70] Gestures, Poses and Actions may be configurable. In one embodiment, possible configurations of Gestures, Poses and Actions are defined in Gesture Configuration Settings. For example, a Gesture Configuration File such as a JSON may define all Gestures, Poses and Actions, along with the available parameters of those Gestures, Poses and Actions. Examples of configurable parameters include:

- pose intensity (what is the weighting on a particular pose)
- gesture intensity (how pronounced or accentuated is the gesture)
- gesture frequency (what is the probability of the gesture being used)

- [71] In one embodiment, Gesture configurations are defined in Gesture Configuration Settings. Gesture Configuration Settings may determine available Gestures and ranges of motions for each type of gesture. Gestures may be “complete” gestures, meaning they include both a complete action and pose, as opposed to being split by pose and action.
- [72] For each Gesture, Configuration Setting may include a range of movements and configurable parameters for that Gesture. For example, acceptable values for the speed of an Action may be restricted between a "speed\_min" and a "speed\_max" value. A gesture speed value may be randomly generated between speed\_min and speed\_max, and provided as input to "[speed,0.98]"
- [73] Gesture frequency defines the probability of a gesture being randomly selected. Each gesture, or category of gestures, may be associated with a frequency. For example, various beat gestures may have the following frequencies: "chop": 0.4, "circle": 0.1, "small\_arc": 0.5, "wide\_arc": 0. When a word has been identified as one that needs a gesture, an appropriate gesture may be selected based on the frequency rates.
- [74] BEAT Action Configuration Settings, for example, for a movement of an arc with palms down, may define a set of available arm poses, wrist poses and hand poses are defined (as some actions are not compatible with some poses). The Configuration Setting also defines amplitude ranges for four preset beat “strengths”, i.e. extra strong, strong, medium, or low. The Emphasis Detection algorithm described herein determines the “strength” of a beat for each word (if any), and the exact strength is randomly chosen within the given range. In runtime, when generating a beat gesture, a random selection may be made from each of the available arm, wrist and hand poses. BEAT pose Configuration Settings may be defined for wrist poses, including variation poses for wrist poses, such as for palms up, palms down, and palms centre.

#### Personality Configuration – Global Configuration Setting

- [75] In one embodiment, Embodied Agents are endowed with different personalities using one or more Global Configuration Settings. Global variables may be set which affect the expression of all Gestures. Global Configuration Settings define the tendency and usage of Gestures within possible ranges. An Embodied Agent’s personality may be configured using Global Configuration Settings.
- [76] In one embodiment, a global Configuration Setting json encapsulates all levers a character author might want to tweak to create a gesturing style: such as Gesture speed, Gesture height and width (average), types of beat action, Hand poses, Wrist orientation, Excitability, hesitancy and any other suitable parameters.

[77] In a further embodiment, the parameters in the global Configuration Setting may be modulated.

[78] In one embodiment, the global Configuration Setting defines the following global parameters:

#### *Speed*

[79] The global Configuration Setting may define parameters that determine the speed of Actions. For example, The global Configuration Setting may determine a minimum speed and a maximum speed for Actions. In one embodiment, different speed parameters may be set for different types of Gestures. For example, symbolic Gestures and beat Gestures may be configured with different speed parameters.

[80] Symbolic Gesture speed defines how fast the Embodied Agent moves into Symbolic Gestures. A minimum speed and a maximum speed for moving into Symbolic Gestures may be defined for the Embodied Agent.

[81] Beat Gesture speed defines how fast the Embodied Agent moves into Beat Gestures. A minimum speed and a maximum speed for moving into Beat Gestures may be defined for the Embodied Agent.

#### *Gesture Type*

[82] rates of different types of beat gestures may be defined. For example:

```
"beat_types": {
  "values": [
    {
      "name": "arc_palm_down",
      "rate": 0.2
    },
    ...
  ]
}
```

#### *Gesture Frequency*

[83] The global Configuration Setting may define the frequency of certain types of Gestures by an Embodied Agent. For example, a maximum number of Symbolic gestures per sentence may be defined, ensuring that the Embodied Agent does not display too many symbolic gestures.

The global Configuration Setting may independently set the rate of strong gestures, medium gestures, and low gestures (which may be used to create variety in Beat Gestures). A weight of 'strong', 'medium' or 'low' is placed on each emphasised word. A global configuration `rate_strong`, `rate_medium`, `rate_low` defines how often gestures of different sizes are used for a personality. The sum of these three values is the overall gesture rate. The global Configuration Setting sets how many strong, medium, and low beats an Embodied Agent utters in a sentence.

- [84] An “emphasis” parameter changes the speed of speech based on the emphasis strength. A negative value will slow down speech. E.g.

```
"emphasis": {
  "tag": "[[speed EMPHASIS]]",
  "strong": -0.25,
  "medium": -0.2,
  "low": -0.15
},
```

- [85] A “head”: configuration adds high-level (#) markup tags on emphasised words based on strength of emphasis and sentiment of the sentence. These high-level tags are defined in a high-level configuration file.

Sentiment\_threshold variables may define the range of neutral sentiment. Sentiment analysis may return a value between -1.0 (full negative) and +1.0 (full positive). Within a type of gesture, the global Configuration Setting may set the frequency of certain subtypes of gestures (e.g. circling actions, chopping actions etc), or even the frequency of individual gestures.

#### *Pose Configuration / Gesture Dimensions*

- [86] The global Configuration Setting may determine the tendencies of gesture dimensions for an Embodied Agent. For example, for Beat Gestures, the global Configuration Setting may define the frequency of different poses, e.g. arm positions. In one embodiment, the global Configuration Setting defines what percentage of an Embodied Agent’s arm positions are in a low, medium or high arm height/position, and independently defines what percentage of an Embodied Agent’s arm positions are in a low, medium or high width from one another. There may be independent configurations for:

- arm\_positions: the rates of different arm heights and widths for beat gestures. height(low, mid, high), width (narrow, middle, wide, extra-wide)
- hand\_positions: the rates of different hand positions/shapes used for beat gestures
- hand\_orientation: Embodied Agent’s tendency to gesture with palms up, centre, or down

#### *Handedness and Symmetry*

- [87] Embodied Agents may be configured to have a “handedness”, by defining the frequency and/or strength of gestures on one hand to be greater than that on the other, in the Configuration Setting.
- [88] The rate of each hand for single-handed symbolic gestures may be defined, e.g.

```
"handedness": {
  "values": [
    {
      "name": "left",
      "rate": 0.5
    }
  ]
}
```

- },  
...  
[89] The rate of non-symbolic (beat) gesturing hands together, vs one or the other may be defined, e.g.

```
"symmetry": {
  "values": [
    {
      "name": "together",
      "rate": 0.4
    },
    ...
  ]
}
```

- Emotion*
- [91] An emotion parameter may define how much the animation of an Embodied Agent is affected by emotion. An emotional\_threshold parameter defines how easily emotion affects an Embodied Agent, by defining how high a sentiment score must be before a size of gesturing is increased. A pose\_speed\_multiplier parameter multiplies the pose speed when the emotional threshold is exceeded. An action\_speed\_multiplier multiplies the action speed when the emotional threshold is exceeded. In other pose and action speed may be modified additively rather than multiplicatively.
- [92] A rate\_multiplier may define how much the Embodied Agent's frequency of gestures increases in response to emotion.
- [93] A size\_level\_offset may increase the size of gestures by a number of levels in response to emotion. A height\_offset may define an increase in the height of gestures, and a hands\_spread\_offset may define an increase in the width of gestures.

#### *Gesture Intervals*

- [94] A gesture\_interval variable may define a minimum and maximum number of words between gestures.
- [95] A first\_gesture\_offset variable may predefine the minimum number of words before the first gesture of a sentence. This ensures that the first gesture doesn't start to play before the Embodied Agent is speaking. That is, that the gesture offset is smaller than the total time the Embodied Agent has been speaking.
- [96] A hesitancy variable may inject hesitancy markers, or filler words (such as "ums" and "ahs").
- [97] The global Configuration Setting may define parameters determining how affected Embodied Agents are by various inputs.
- [98] For example, emotional modulation may be achieved by setting a variable which determines how affected an Embodied Agent is from the sentiment of a sentence.

- [99] However, sentence sentiment is only one example of input which may affect the behaviour of the embodied agent. Other aspects may include audio input (e.g. from the agent's virtual environment or from a user via a microphone), visual input (e.g. from the agent's virtual environment or from a user via a camera), input from a user interface, or any other suitable input.
- [100] The parameters within the global Configuration Setting may be associated with multipliers, which are set using modulatory rules. For example, action speed multipliers may be set to modulate the speed of gesturing, and rate multipliers may modulate the frequency of gestures. A size level offset may increase the amplitude of gestures (resulting in gestures getting "bigger" or "smaller").

#### *Randomization*

- [101] By defining ranges of gesture parameters, and frequencies of gestures, the global Configuration Setting parameters affect the degree of variation and randomization of autonomous animation.

#### *Modulation*

- [102] At step 106, Modulation may include:
- swapping out animation files (so that one individual uses eg "wave01" and another uses "wave02" in the same place in speech);
  - using different gestures (so one individual uses "chop" and another "circle" for emphasis);
  - increasing or decreasing speed or amplitude of gestures (S);
  - modifying the rate of gesturing (how many gestures Embodied Agents carry out).
  - Modulation may modify the overall rates of gesturing, and/or rates of certain types of gesturing. Rates of gesturing can be set in a Configuration Settings, and determines how many gestures (of various kinds) are applied to sentences.

A modulation Module may modify and/or be modified by clause analysis and/or markup generation.

- [103] Demographic Modulation creates differences in the gesturing style of Embodied Agents across factors like age, gender, race, and culture. For example, Embodied Agents portraying younger characters may be more expressive and less dominant than older characters. Some gestures are meaningful only within a specific culture or may have quite different meanings in different cultures (even when they speak the same language).
- [104] Personality Modulation may modulate gestures to align with personality traits such as extroversion, introversion, confidence, friendliness, openness. These are defined in a config and map onto more fine-grained behavioural traits (eg high energy). The fine-grained traits map onto low-level

differences in gesture mark-ups (eg more frequent, bigger, faster gestures). These differences are implemented by using different average values for gesturing rate, amplitude, and speed respectively. Further examples of personality modulation include: higher rates of embodiment gestures for nervous or less confident personalities (these are inserted between clauses with some probability – change the probability to change how many they do on average); wider variety of gestures for more expressive personalities (set the rates of each gesture to be lower but greater than zero for many gestures, vs higher rates for a smaller number of different gestures); higher prevalence of palms-up, open hand, more fluid/smooth arcing gestures for friendlier and more open personalities; higher prevalence of rigid pontificating gestures for more authoritative personalities (set a higher rate for eg gestures in which the palms are up).

- [105] Style Modulation may apply idiosyncratic gesturing styles to Embodied Agents. Style Modulation may be more fine-grained than personality modulation, and define low-level gesture characteristics, such as whether an Embodied Agent tends to gesture with a relaxed palm up hand pose, or a stiff fingers spread palm down hand pose (or many other options), and whether they tend to use chop actions, circling actions, fluid arcing actions etc, and whether they tend to use their left or right hand, or tend to gesture symmetrically. All of these can be defined broadly by their personality, but they can be tweaked to give the individual character a unique style. These are all defined in a high-level/personality Configuration Settings, in which the rate of left/right/both hands can be set, and the rate of chop gestures and circling gestures, etc.
- [106] Role Modulation enables a single Embodied Agent to display different gesturing behaviour depending on the role they are in at the time, even for the same utterance. For example, if a person is presenting an idea at a conference talk, they will likely use different gestures to when they are engaging in a casual conversation, even if in both cases they're saying the same words. Other roles may include explaining or outlining some facts, guiding or advising, tutoring or teaching. The particular role that the character is playing interacts with their personality and idiosyncratic style to form the resulting overall gesturing style.
- [107] Sentiment Modulation refers to using the results of sentiment analysis to trigger specific gestures, and also to modulate potentially any or all other gestures. The specific gestures might be smiles and eyebrow raises, thumbs up or clapping for pleased or happy emotions, especially for expressing pleasant surprise, or frowns and clenched fists for expressing anger or frustration. The arousal expressed in the clause also modulates the gestures that are chosen. For example, high arousal (such as clauses expressing excitement or frustration) will mean that the poses (the starting points of the

gestures) become wider and higher, fingers become more spread, gestures become more frequent, and actions become bigger and faster.

- [108] This is achieved in two ways: first, by adding offset values to the frequency of gestures and the amplitude and speed of each gesture. The offset is positive for high arousal, and negative for low arousal, and is scaled so that the higher the arousal, the higher the offset and vice versa.
- [109] Second, for the arm and hand poses, a variation pose is blended in. For the arms, the variation pose is the widest and highest pose (for high arousal), which is blended with the base pose to a small-medium degree to ‘pull’ the base pose for each gesture wider and higher. For the hands, the variation pose is the fingers at maximal spread blended to a small-medium degree, which pull the fingers slightly more spread in whichever base pose they are in. These offsets and degrees of variation poses are configurable as part of the modulation of personality and gesturing style. For example, one character may be more expressive than another, so highly emotional content will have a larger impact on their gesturing behaviour.
- [110] Sentence-level emotion configuration takes the overall sentiment of a sentence and applies the relevant change in emotion. Each emotion (such as anger, concern, disgust, fear) may be connected to a dictionary (defining words triggering the emotion). For each emotion, low, mid and high values of the emotion may be defined, each having an intensity and a duration. The intensity of the detected emotion may be determined by sentiment analysis. A duration may define how long the emotion lasts. An intensity multiplier define the extent to which a base emotion is negated.
- [111] The Agent may be simulated using a neurobehavioral model (biologically modelled “brain” or nervous system), comprising a plurality of modules having coupled computational and graphical elements. Each module represents a biological process and includes a computational element relating to and simulating the biological process and a graphical element visualizing the biological process. Thus, the Agent may be “self- animated” to perform certain behaviour without external control and thus exhibit naturally occurring automatic behaviour such as breathing, blinking, looking around, yawning, moving its lips. Biologically based autonomous animation may be achieved by modelling multiple aspects of the nervous system, including, but not limited to, the sensory and motor systems, reflexes, perception, emotion and modulatory systems, attention, learning and memory, rewards, decision making, and goals. The use of a neurobehavioral model to animate a virtual object or digital entity is further disclosed in: Sagar, M., Seymour, M. & Henderson, A. (2016) Creating connection with autonomous facial animation. Communications of the ACM, 59(12), 82-91 and

WO2015016723A1, also assigned to the assignee of the present invention and is incorporated by reference herein.

- [112] The Autonomous Animation System may give and receive signals to and from the neurobehavioural model. Sending signals allows the sentiment and content of the Embodied Agent's utterances to affect their internal emotional state, which in turn may affect their underlying emotional or idle animations. Receiving signals allows external factors to affect their gestures, such as the character's perception of the user's emotional state or identification of objects in the field of view, allowing them to be more responsive to the user and the situation. Another example is detecting that the user is paying attention and if not, introduce some speech disfluency: for example, stopping and restarting clauses.

### **Variation Poses**

- [113] Instead of adding random variation to each particular joint (which may result in unnatural poses), a Variation Pose system enables the blending between two or more coherent Input Poses to create a new pose Variation Pose. Input Poses may be deliberately authored by an animator to blend in a coherent manner.
- [114] Figure 6 shows blending between arm Variation Poses. Figure 6A shows an Input Pose of a wide stance, Figure 6B shows a Variation Pose configured to blend with the pose of Figure 6A. Figure 6C shows a Blended Pose which is an intermediate pose between Figure 6A and Figure 6B.
- [115] Figure 7 shows a first example of blending between hand Variation Poses. Figure 7A shows an Input Pose of an outstretched hand, Figure 7B shows a Variation Pose, of a folded hand, configured to blend with the pose of Figure 7A. Figure 7C shows a Blended Pose which is an intermediate pose between Figure 7A and Figure 7B.
- [116] Figure 8 shows a second example of blending between hand Variation Poses. Figure 8A shows an Input Pose of a hand with curled fingers, Figure 8B shows a Variation Pose configured to blend with the pose of Figure 8A. Figure 8C shows a Blended Pose which is an intermediate pose between Figure 8A and Figure 8B.
- [117] In one embodiment, the TTG System generates a Variation Pose using the following steps:
- Select or receive an Input Pose. In one embodiment, the Input Pose is a "base pose", which means it is the default pose in which a body part of the Embodied Agent is configured.
  - Select or receive a corresponding Variation Pose, configured to blend with the Input Pose.
  - Blend between each Input Pose and one or more Variation Poses to generate a Blended Pose.

- [118] In one embodiment, an Input Pose and the Variation Pose are each selected with an intensity, and blended together (e.g. 0.8 Pose 1 is blended with 0.9 Pose 2).
- [119] In another embodiment, two or more Variation Poses, configured to blend with one another are selected, and blending weights between each of the poses is randomly generated, specifying the degree to which the Variation Poses are blended (e.g. 0.2 Pose1 is blended with 0.4 Pose2 and 0.4 Pose3).
- [120] Poses selections may be restricted to be compatible with the action that is about to come. There may be predefined a set of compatible poses for each action from which one is chosen.

### **Autonomously Emotive Speech**

- [1] In one embodiment, Embodied Agents are autonomous dynamic systems, with self-driven behaviour, which can also be controlled (in a weighted or controllable fashion) externally by the TTG System as described herein, allowing a blend of autonomy (wherein Embodied Agent gestures are driven by their internal emotional states) and directability (wherein Embodied Agent gestures are driven by text as per the TTG System). “Bottom up” autonomous behaviour may be facilitated by a programming environment such as that described in the patent US10181213B2 titled “System for Neurobehavioural Animation”. A plurality of Modules are arranged in a required structure and each module has at least one Variable and is associated with at least one Connector. The connectors link variables between modules across the structure, and the modules together provide a neurobehavioral model. Variables and/or Modules may represent neurotransmitter/neuromodulators such as dopamine or oxytocin, which may be used to affect the operation of the structure.
- [2] The neurobehavioural model may include an emotional system as described in the patent application PCT/IB2020/056280, ARCHITECTURE, SYSTEM, AND METHOD FOR SIMULATING DYNAMICS BETWEEN EMOTIONAL STATES OR BEHAVIOR FOR A MAMMAL MODEL AND ARTIFICIAL NERVOUS SYSTEM, incorporated by reference herein.
- [3] For each word carrying emotional content, the TTG System may output both a possible gesture plus one or more emotional impulses. Each emotional impulse perturbs the state of the internal emotional system. The internal emotional system is a dynamical system in flux, with emotions competing against each other and sustaining and decaying, providing a history of emotional states.
- [4] Thus, the internal emotional reaction of the Embodied Agent depends on the content and order or sequence of the word.
- [5] In one embodiment, the TTG System may process each word sequentially and output one or more emotional impulses as soon as the word is processed. In another embodiment, the TTG System may

process an entire clause, sentence, and/or paragraph, and output emotional impulses according to any suitable rules or analysis of the sentence.

- [6] Thus, the Autonomously Emotive Speech drives the emotional system in a layerable, blendable way with history - by the content of Input Text (e.g. key words or sentiments) affecting the internal state of the Embodied Agent so that emotions linger, and blend appropriately.
- [7] In one embodiment, words may be decomposed into two or more underlying emotions. For example, the word "marvellous" can be both construed as "surprising" and "happy", and "horrified" can be decomposed into "fear" + "disgust". In one embodiment two or more "emotion dictionaries" each contain lists of words representing elements of a particular emotion. Words or tokens are matched against the emotion dictionaries to determine which component emotions apply to the words or tokens.
- [8] In one embodiment, each word matched in an emotion dictionary may also be paired with a dictionary match variable representing the degree to which the word is relevant to the emotional dictionary. For example, a "fear" dictionary may contain words with corresponding dictionary match variables as follows: horrifying 0.9, disaster 0.92, scary 0.8, uncomfortable 0.6. Both the matched emotions as well as dictionary match variables may be returned and provided as input to the emotion system. This provides a way of responding to complex, compound emotions in a compositional, blendable and transitional way.

### **Emphasis Detection**

- [9] An Emphasis Detection algorithm determines the importance of words in a Communicative Utterance, enabling an Embodied Agent to emphasise the most important words with gestures. A Emphasis Detection Algorithm may identify key words according to certain criteria. In one embodiment, the Emphasis Detection Algorithm identifies which words in each clause will be given a strong, medium, low, or no emphasis.
- [10] Figure 2 shows a Emphasis Detection algorithm according to one embodiment. At step 202, an Input Text is received. At step 204, for each "token" or word  $w$  in the Input Text, each Emphasis Detection rule is applied. Calculation of the word score may include the application of several rules. At step 206, for each Emphasis Detection Rule, a rule score is calculated for the relevant token or word. Emphasis Detection rules may be weighted such that some rules have greater influence on the word score than others. At step 208, an overall Emphasis Score for the token or word is calculated. At step 210, the Emphasis Scores for each rule are returned. The Emphasis Scores for the words are then used to apply Gestures based on the Emphasis Scores

- [11] In one embodiment, the Emphasis Detection algorithm looks up the rareness of each word. A look-up table of words and associated “frequencies” (of use of that word in a particular language or context) may be used to return word rareness for each word.
- [12] Words with relatively higher Emphasis Scores may “trigger a “beat”, which is a type of gesture which does not carry any speech content, but conveys non-narrative content and aligns with the rhythm of speech. The Emphasis Detection recognises the parameters in which the keyword has been defined to activate rules.
- [13] A “weight” or intensity may range between of 0-1. Weights are specified for each rule. Weights may be applied in two ways: “weight” per rule and “weight” per word.
- [14] The weight of the rule remains a constant e.g. the sentiment rule is always weighted at a value of 0.8. Meanwhile a keyword will be weighted depending on its stated value within the corresponding dictionary e.g. I am very excited (listed as 0.7 in the sentiment dictionary).
- [15] Multiple keywords may be identified in a given sentence and emphasized with beat gestures accordingly. In one embodiment, the Emphasis Detection algorithm identifies keywords in a given clause, and assigns all words high, medium, low or no emphasis based on the weighted keyword identification algorithm. Scores are calculated for all words in a sentence, then sorted in descending order. The top 10% are defined as strong beats, following 10% as medium beats, another following 10% as low beats. Any suitable thresholds may be provided to categorize beats as strong, medium and/or low.
- [16] Beat Gestures may be applied to the stressed syllable such that the stroke of the beat is in sync with the stressed syllable in a word.
- [17] Rules may be combined in any suitable manner, including summing or finding the MAX. One example of suitable rules weightings is shown in Figure 3. Figure 4 shows an example of the application of rules to the input text “John loves snorkelling in Greece”.

#### *Emphasis Detection Fine-Tuning*

- [18] The weights for the Emphasis Detection rules may be fine-tuned using, for example, a greedy algorithm or a deep learning model, on human-annotated data. A collection of sentences (preferably over 1500), covering various semantic domains are selected as a training dataset. Human annotators manually select the keywords (emphasis words) for each sentence. In total 3540 sentences are used as training dataset. In one embodiment, a plurality of annotators are used, and the conformity of their annotation decisions may be measured. In one experiment, the applicants found that two human annotators agreed on 71.44% of

emphasized words. The annotations from all annotators may be used at the same time to avoid overfitting to single annotation.

- [19] In one embodiment, the weights are fine-tuned using a greedy algorithm. A greedy algorithm is used to tweak the weights to obtain maximum accuracy on training data. All weights are initialised randomly. At each iteration, all weights are fixed except for one randomly chosen. It will be tuned by searching in a 0.01 precision within [0,1] to maximize the accuracy of training data. The algorithm terminates after 10k iterations.
- [20] In another embodiment, a deep neural network is used to train the weights. A 1-layer fully connected feedforward network without bias or activation is used from Keras to find the weights.

### **Advantages**

- [21] The TTG System creates impressions of different personalities by varying the gesturing style of a Embodied Agent. The TTG System is highly configurable. A person with an understanding of personality and body language, for example a film director, can use this system to create different realistic behaviours in Embodied Agents. The person can choose the set of gestures used, for example palm up vs palm down. They can also adjust the speed, rates, size and location of their gesturing. They can specify how emotionally expressive the agent is, by configuring how the gestures get affected by the sentiment of the sentence. All of the above aspects influence the perceived personality of the agent.
- [22] An Action and Pose scheme is used to generate a large variety of gestures efficiently, in a manner requiring less computational storage space. The Action and Pose scheme also saves animator time as a large set of animations may be generated automatically using the Action and Pose scheme without requiring all variations to be manually crafted by an animator.
- [23] The system identifies the gesture types most commonly used in dialogs, including:
- Symbolic gestures (iconic, metaphoric, emblematic) - Identified based on string-matching and dictionaries. E.g., tracing a square for the word “square”; using up gesture for “higher”.
  - Dialogue Act gestures – Identified by our rules based on Linguistics. E.g., small shrug and open palm arc outward for question; head shake and dismissive flick of wrist for negation; pointing left and then right on “this or that” in “you can have this or that”
  - Emphasising gestures – Identified using keywords detection. E.g., applying a beat gesture to “really” in “this is really bad”
  - Embodiment gestures - E.g., looking up and to one side and furrowing brow and then looking back as if to be retrieving the term “constructivist epistemology”; shifting weight from one foot to the other between clauses

- Turn-taking gestures - E.g., looking away between clauses when not finished (retaining conversational floor), looking directly at user and smiling when finished (ceding conversational floor)
- [24] The TTG System results in more human-like autonomous animation because the TTG System derives linguistic information from Input Text which helps inform animation. The TTG System detects negations based on the relationships between words in the dependency tree that represents the sentence. The TTG System detects enumerating behaviours by finding noun phrases, verb phrases, and other patterns in the part of speech of words.
- [25] Variation Poses introduce natural looking randomness to the gestures.

## INTERPRETATION

- [26] The methods and systems described may be utilised on any suitable electronic computing system. According to the embodiments described below, an electronic computing system utilises the methodology of the invention using various modules and engines. The electronic computing system may include at least one processor, one or more memory devices or an interface for connection to one or more memory devices, input and output interfaces for connection to external devices in order to enable the system to receive and operate upon instructions from one or more users or external systems, a data bus for internal and external communications between the various components, and a suitable power supply. Further, the electronic computing system may include one or more communication devices (wired or wireless) for communicating with external and internal devices, and one or more input/output devices, such as a display, pointing device, keyboard or printing device. The processor is arranged to perform the steps of a program stored as program instructions within the memory device. The program instructions enable the various methods of performing the invention as described herein to be performed. The program instructions, may be developed or implemented using any suitable software programming language and toolkit, such as, for example, a C-based language and compiler. Further, the program instructions may be stored in any suitable manner such that they can be transferred to the memory device or read by the processor, such as, for example, being stored on a computer readable medium. The computer readable medium may be any suitable medium for tangibly storing the program instructions, such as, for example, solid state memory, magnetic tape, a compact disc (CD-ROM or CD-R/W), memory card, flash memory, optical disc, magnetic disc or any other suitable computer readable medium. The electronic computing system is arranged to be in communication with data storage systems or devices (for example, external data storage systems or devices) in order to retrieve the relevant data. It will be understood that the system herein described includes one or more elements that are arranged to perform the various functions and methods as described herein. The embodiments herein described are aimed at providing the reader with

examples of how various modules and/or engines that make up the elements of the system may be interconnected to enable the functions to be implemented. Further, the embodiments of the description explain, in system related detail, how the steps of the herein described method may be performed. The conceptual diagrams are provided to indicate to the reader how the various data elements are processed at different stages by the various different modules and/or engines. It will be understood that the arrangement and construction of the modules or engines may be adapted accordingly depending on system and user requirements so that various functions may be performed by different modules or engines to those described herein, and that certain modules or engines may be combined into single modules or engines. It will be understood that the modules and/or engines described may be implemented and provided with instructions using any suitable form of technology. For example, the modules or engines may be implemented or created using any suitable software code written in any suitable language, where the code is then compiled to produce an executable program that may be run on any suitable computing system. Alternatively, or in conjunction with the executable program, the modules or engines may be implemented using, any suitable mixture of hardware, firmware and software. For example, portions of the modules may be implemented using an application specific integrated circuit (ASIC), a system-on-a-chip (SoC), field programmable gate arrays (FPGA) or any other suitable adaptable or programmable processing device. The methods described herein may be implemented using a general-purpose computing system specifically programmed to perform the described steps. Alternatively, the methods described herein may be implemented using a specific electronic computer system such as a data sorting and visualisation computer, a database query computer, a graphical analysis computer, a data analysis computer, a manufacturing data analysis computer, a business intelligence computer, an artificial intelligence computer system etc., where the computer has been specifically adapted to perform the described steps on specific data captured from an environment associated with a particular field.

## CLAIMS

1. A method of animating a virtual character or digital entity, including the steps of:
  - i. receiving Input Text specifying words to be spoken by the virtual character or digital entity;
  - ii. determining a Pose to be applied to Input Text;
  - iii. determining an Action to be applied to Input Text and
  - iv. generating at least one motion of the virtual character or digital entity representing the Action being applied from the Pose.
2. The method of claim 1 wherein the pose is an arm pose and the method of determining a pose includes the step of determining a horizontal distance between arms.
3. The method of claim 1 or claim 2 wherein the pose is an arm pose and the method of determining a pose includes the step of determining a vertical height of one or more arms.
4. The method of any one of claims 1 to 3 wherein the at least one motion is a beat gesture.
5. A system for animating a virtual character or digital entity, the system including:
  - i. an input-module, the input module receiving input text;
  - ii. a determination module, the determination module determining:
    - i. a pose to be applied to Input Text; and
    - ii. an Action to be applied from the at least one Pose; and
  - iii. an output module, the output module generating at least one motion of the virtual character or digital entity based on the Pose and the Action.
6. A method of animating a virtual character or digital entity, including the steps of:
  - i. determining an Input Pose to be animated;
  - ii. determining a Variation Pose representative of a Gesture, the Variation Pose configured to blend with the Input Pose;

- iii. determining a Blended Pose comprising a weighted interpolation between the Input Pose and the Variation Pose; and
  - iv. animating the virtual character or digital entity using the Blended Pose.
7. The method of claim 1 wherein the step of determining a pose includes the steps of the method of claim 6.
8. A system for animating a virtual character or digital entity, the system including:
- i. a determination module, the determination module determining:
    - i. an Input Pose to be animated;
    - ii. a Variation Pose representative of a Gesture, the Variation Pose configured to blend with the Input Pose;
    - iii. a Blended Pose comprising a weighted interpolation between the Input Pose and the Variation Pose; and
  - ii. an animating module animating the virtual character or digital entity using the Blended Pose.
9. A method of animating a virtual character or digital entity, the virtual character or digital entity having at least one arm, wrist, and digit, the method including the steps of:
- i. determining at least two of:
    - an arm pose selected from a range of arm poses;
    - a wrist pose selected from a range of wrist poses; and
    - a digit pose selected from a range of digit poses; and
  - ii. animating the virtual character or digital entity to display an overall pose including the arm pose, the wrist post, and/or the digit pose.

10. The method of claim 9 wherein determining at least two of is determined randomly.
11. A method of animating a virtual character or digital entity, the virtual character or digital entity having at least one limb, the method including the steps of:
  - i. determining at least:
    - a limb pose selected from a range of limb poses; and
    - a pose of a sub part of the limb selected from a range of sub part poses; and
  - ii. animating the virtual character or digital entity to display an overall pose including the limb pose and the pose of a sub part of the limb.
12. The method of claim 11 wherein determining at least is determined randomly.
13. A method of animating a virtual character or digital entity, including the steps of:
  - i. receiving Input Text specifying words to be spoken by the virtual character or digital entity;
  - ii. determining an emphasis score of each word in the Input Text;
  - iii. determining a set of words with relatively higher emphasis score compared to the remaining words in the Input Text; and
  - iv. animating a virtual character or digital entity to speak the Input Text and applying a gesture to each word from the set of words with relatively higher emphasis score.
14. The method of claim 13 wherein the gesture is applied to a stressed syllable of each word from the set of words with relatively higher emphasis score.
15. The method of claim 13 of claim 14 wherein the gesture is a beat gesture.
16. The method of any one of claims 13 to 15 wherein the emphasis score is based on word rarity, wherein words with higher rarity have a higher emphasis score.
17. The method of any one of claims 13 to 16 wherein the set of words with relatively higher emphasis score comprises words from the Input Text with an emphasis score within a predefined top percentile of all words in the Input Text.

18. The method of any one of claims 13 to 16 wherein the gesture applied to each word from the set of words has a gesture amplitude proportional or substantially proportional to the emphasis score of the word.
19. The method of any one of claims 13 to 16 wherein the emphasis score is calculated by applying a set of criteria to determine the emphasis score of each word, wherein the contribution of each criterion to the emphasis score is weighted using a weighting.
20. The method of claim 19, wherein the set of criteria are selected from the group consisting of: word sentiment, part of speech, capitalization, negation and rarity.
21. The method of claim 20 wherein the weighting is optimized to emphasize words in the same manner as a human annotator.
22. The method of claim 21 wherein the weighting is optimised using machine learning.
23. A system for animating a virtual character or digital entity, the system including:
  - i. input-receiving means for receiving input text;
  - ii. a plurality of gestures, wherein each gesture is associated with:
    - i. an animation;
    - ii. at least one configurable parameter for varying the animation; and
    - iii. a configuration range of the configurable parameter;
  - iii. an animation generator, wherein the animation generator is configured to:
    - i. analyse input text to determine at least one gesture;
    - ii. determine a configuration of the configurable parameter from the configuration range; and
    - iii. animate the virtual character or digital entity with the animation as varied by the configurable parameter.
24. The system of claim 23 wherein each gesture is associated with at least one modulatory variable for modulating the configuration range of the configuration parameter, and wherein

the animation generator is configured to determine a configuration of the configurable parameter as modified by the modulatory variable.

25. The system of claim 23 or claim 24 wherein determining a configuration of the configurable parameter from the configuration range is randomly determined.
26. The system of any one of claims 23 to 25 wherein configurable parameters are selected from the group consisting of: gesture speed, gesture amplitude and gesture pose.
27. A method of animating a virtual character or digital entity, including the steps of:
  - i. configuring one or more parameters of a set of gestures, the parameters being configured such that the gestures reflect traits of the virtual character or digital entity; and
  - ii. animating the virtual character or digital entity with the gestures as configured by the one or more parameters.
28. The method of claim 27 wherein the traits include one or more of culture, gender, style, personality or character.
29. The method of claim 27 or claim 28 wherein the parameters are selected from one of more of the group of gestures comprising:
  - i. the speed of gestures;
  - ii. the position of gestures;
  - iii. the frequency of gestures;
  - iv. the amplitude of gestures; and
  - v. the handedness of gestures.
30. A method of animating a virtual character or digital entity including the steps of:
  - i. receiving Input Text specifying words to be spoken by the virtual character or digital entity;
  - ii. analysing the Input Text to determine at least one gesture type, wherein each gesture type includes a plurality of gestures available for selection and an associated frequency value;

- iii. selecting gestures to be applied to Input Text using gesture frequency values; and
- iv. animating the virtual character or digital entity using the selected gestures.

31. A method of animating a virtual character or digital entity, including the steps of:

- i. receiving Input Text specifying words to be spoken by the virtual character or digital entity;
- ii. generating a Parse Tree ;
- iii. using the Parse Tree to determine at least one animation of the virtual character or digital entity; and
- iv. generating at least one motion of the virtual character or digital entity based at least on the animation.

32. The method of claim 31 wherein the parse tree is a dependency tree.

33. The method of claim 31 or claim 32 further including the step of using the Parse Tree to determine a scope of a negation, and using the scope of the negation to determine the animation.

34. The method of any one of claims 31 to 33 further including the step of using the Parse Tree to detect enumeration, and using the enumeration to determine the at least one animation.

35. The method of any one of claims 31 to 34 further including the step generating mark-up representative of the at least one animation, and using the mark-up generate the at least one motion of the virtual character or digital entity.

36. The method any one of claim 31 to 35 wherein the at least one animation is facial and/or body animation.