



## Warping character animations using visual motion features

Alberto Jovane, Pierre Raimbaud, Katja Zibrek, Claudio Pacchierotti, Marc Christie, Ludovic Hoyet, Anne-Hélène Olivier, Julien Pettré

### ► To cite this version:

Alberto Jovane, Pierre Raimbaud, Katja Zibrek, Claudio Pacchierotti, Marc Christie, et al.. Warping character animations using visual motion features. Computers and Graphics, 2022, 10.1016/j.cag.2022.11.008 . hal-03877972v1

**HAL Id: hal-03877972**

**<https://inria.hal.science/hal-03877972v1>**

Submitted on 1 Dec 2022 (v1), last revised 30 Mar 2023 (v2)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



# Warping Character Animations using Visual Motion Features <sup>★</sup>

Alberto Jovane<sup>a,\*</sup>, Pierre Raimbaud<sup>a</sup>, Katja Zibrek<sup>a</sup>, Claudio Pacchierotti<sup>a</sup>, Marc Christie<sup>a</sup>, Ludovic Hoyet<sup>a</sup>, Anne-Hélène Olivier<sup>a</sup>, Julien Pettré<sup>a,\*</sup>

<sup>a</sup>Inria, University of Rennes, CNRS, IRISA, France

## ARTICLE INFO

### Article history:

Received 24 November 2022

**Keywords:** Computers and Graphics,  
Formatting, Guidelines

## ABSTRACT

Despite the wide availability of large motion databases, and recent advances in exploiting them to create tailored contents, artists still spend tremendous efforts in editing and adapting existing character animations to new contexts. This motion editing problem is a well-addressed research topic. Yet only a few approaches have considered the influence of the camera angle, and the resulting visual features it yields, as a mean to control and warp a character animation. This paper proposes the design of viewpoint-dependent motion warping units that perform subtle updates on animations through the specification of visual motion features such as visibility, or spatial extent. We express the problem as a specific case of visual servoing, where the warping of a given character motion is regulated by a number of visual features to enforce. Results demonstrate the relevance of the approach for different motion editing tasks and its potential to empower virtual characters with attention-aware communication abilities.

© 2022 Elsevier B.V. All rights reserved.

## 1. Introduction

Numerous interactive applications, *e.g.* video games or Virtual Reality immersive simulations, rely on motion capture techniques to animate human characters thanks to the excellent trade-off they provide between computational budget and animation quality [9]. To extend the usability of such motion captured animations beyond what was initially recorded, a long-standing objective in the field of computer animation is to design efficient techniques to edit motion clips, originally taking inspiration from signal processing editing approaches [12]. Motion editing techniques act on various features of animation trajectories, *e.g.*, blending similar motions to generate new ones [51], concatenating motion clips [6] or accounting for in-

teractions between characters during concatenation [52]. Motion style transfer has also been addressed [10], as well as adjusting kinematic constraints from geometric relationships in changing environments [3, 23].

This topic is a long-standing one – we voluntarily picked some of the oldest and seminal papers – and a very large body of literature can be found on each of the various facets of motion editing [29, 54]. It is however noticeable that existing techniques rarely consider that human movement is often performed in relation to an observer. This is particularly true when motion is considered as a nonverbal communication channel [31]. Indeed, in such communication tasks, humans control their movements by fundamentally taking into account how it can be visually perceived by an observer. For instance, while waving at someone – a typical voluntary nonverbal communication gesture to attract attention – one makes sure his/her hand is visible to this person, *e.g.* adjusting body orientation, waving amplitude and speed to make the motion salient enough, as well as moving his/her face and eyes to enable gaze contact and ensure that attention is successfully attracted. This example clearly

<sup>★</sup>This is the author version, the final edited document is available at <https://dx.doi.org/10.1016/j.cag.2022.11.008>

\*Corresponding authors  
*e-mail:* [alberto.jovane@inria.fr](mailto:alberto.jovane@inria.fr) (Alberto Jovane),  
[julien.petre@inria.fr](mailto:julien.petre@inria.fr) (Julien Pettré)

highlights the links between the kinematics of a motion and the visual features perceived by an observer. This notion of *visual motion features* are defined in relation with an observer's point of view and Field of Vision (FoV). It covers features such as visibility and centrality of limbs, coverage of limbs and of their motion, generated optical flow, motion amplitude in the view plane, etc.

As visual features of motions can be of prime importance, the objective of this paper is to devise a motion editing technique that enables controlling visual motion features of a given character relatively to an observer. In other words, given a character motion  $m$ , the corresponding set of visual features  $s$  as perceived by an observer, and the *desired* visual motion feature values  $s^*$ , we search for the warping operation  $w$  to synthesise the warped motion  $m_w$  with the desired  $s^*$  values. We view the relevance of our visual feature approach when animating characters in social interactions, or in immersive Virtual Reality applications where characters are meant to adjust themselves to the users' state in the virtual scene.

Motions can be synthesised from the analysis of visual properties. We see examples in animation, *e.g.* where images of the environment are exploited to generate character's movements [14], as well as in robotics, *e.g.* where camera feedback can guide the kinematics's properties of the motion of robotics's limbs. Indeed, our approach is inspired by robotics visual servoing [15], a technique to control a robot's motion based on visual sensor feedback, *e.g.* a camera. In analogy with visual servoing, we want to control a character motion with respect to an observer position. Following this analogy, our motion warping goal is formulated as a visual task, *i.e.* to reach  $s^*$ . Unlike previous approaches where motion editing goals generally rely on the kinematic properties of the motion, we directly set the visual properties we want for the edited motion. However, a major difference with visual servoing solutions is that we want to perform motion warping, not full motion control.

Our main contribution is a new visually driven motion editing framework that enables to manipulate the visual features of an existing motion through an observer's viewpoint. We believe that such visual editing techniques will help enhanced awareness and reactivity in interactive virtual agents, thus we present a user study in Virtual Reality where we demonstrate the advantages of our approach compared to a standard technique. Additionally, we will present a set of use cases to illustrate how our technique can adapt existing content (motion capture) to new contexts, and may empower virtual agents with the ability to capture attention.

## 2. State of the art

In the large body of literature about motion capture editing, our work falls in this category of solutions that can synthesise, control or adjust data-driven character animations on the fly. Different editing aspects were explored in the field such as concatenating new motion sequences from differently ordered ones, *e.g.* motion graphs, motion fields or motion matching techniques [5, 44, 3, 43, 45, 34], interpolating example motions of the same kind to control some kinematic

features [51, 42, 48], performing temporal adjustments of motion (*e.g.* time-warping to align motions before transiting or blending) [6, 24, 1, 57, 53], retargeting motions to new characters of various morphology [25, 17, 55], satisfying kinematic constraints when played in new environments (*e.g.* contact displacement) [56, 44, 32, 2], performing stylistic variations [10], maintaining physical interactions between characters while editing [52, 38, 46].

Even though all these solutions would play on character motion properties and on their visual properties, we are not aware of existing work trying to directly provide control over visual motion features as we aim to do. The closest contributions mainly propose through-the-lens techniques, with which trajectories in the image space, such as hand-drawn sketches [28] or spatial-temporal constraints [18], enable the control of kinematic chains. A similar sketch-based approach was proposed to warp existing animations using motion trails [16]. Solutions rely on constrained optimisation, minimizing a projected difference with the input sketches, while respecting the original animation and disambiguating depth. The perceptual effect of edited animations was also explored to assess the loss in animation quality [4]. Some works in virtual cinematography [37] study the complementary problem of optimizing the point of vision of a virtual scene to respect visual constraints over the motion of the virtual actors.

Establishing relations between motion variables and motion visual features can however be an interesting solution in the frame of nonverbal communication situations where vision is a major perception channel. Nonverbal communication refers to all modes of communication that do not use speech. This includes facial expression, para-linguistic, proxemics and touch, as well as body gestures and postures [30]. During social interactions, a continuous exchange of such signals is possible, mainly because people are able to see each other [13]. This means that the interpretation of social signals involving a mutual interaction between the observer and the other(s) person(s) should be expressed in the reference frame of the observer, namely his/her FoV. This visual perception for social interaction then enables to infer on a person's intention [8, 41], personality [47], as well as emotions [19, 50].

In this context, authors have proposed several approaches to provide animated characters with personality or emotion, such as Durupinar *et al.* [21] who used Laban Motion Analysis, or Randhavane *et al.* [49] who played on the kinematic parameters of gait and gaze. Elaborate solutions for style transfer were also proposed [33], where style can encompass aspects of personality or emotional state. But, it is important to note that these methods are not designed to express these features in relation to the FoV of the observer. A visual-optimization approach is however presented by Huang and Kallmann [36], where character motion, posture and location are synthesised in relation to an observer. Despite presenting a comprehensive framework and sharing similar motivation than our approach, their work is however limited to a set of predefined actions and only considers visibility constraints for character placement. We believe that our approach extends this character-observer relation to general motions and directly links posture adaptation with

vision-based features.

To conclude, to our knowledge, there is no previous attempt to link low-level motion control variables to perceived visual features with the intent to control them, which is the objective set for our work.

### 3. Method

In this Section, we describe the mathematical formulation of our approach. Section 3.1 introduces the main concepts, Section 3.2 presents the different visual motion features we consider when warping the animations, Section 3.3 details the design of our warping operators, and Section 3.4 explains the regulation of the visual motion features using our warping operators.

#### 3.1. Overview

The aim of our approach is to provide designers or interactive applications with a warping controller that adjusts a character animation using view-dependent visual motion features. The input of our system is (i) an animated character, (ii) a set of designer-specified visual motion features to fulfill (*e.g.* visual coverage, vertical extension), and (iii) a camera angle or trajectory that views the character in its environment. Our system adjusts the character animation on a per-frame basis by satisfying user-defined visual motion features in an inverse design approach, from features to parameters.

Technically, we propose to express the problem as a specific instance of the eye-to-hand visual servoing principle [20] in which a camera, fixed or animated in the world, observes the motion of one or multiple kinematic chains. Rather than driving the velocity of kinematic chains from on-screen velocities – as implemented in traditional visual servoing tasks [22] or with through-the-lens control [26], our objective is to develop a control law that updates velocities in the kinematic chain by regulating the difference between globally measured visual motion features and expected ones in the 2D camera space. This requires the design of (i) **estimators** that are able to measure the values of the expected visual motion features from the given camera, (ii) **warping units** that alter the parameters of the kinematic chains, and (iii) a **control loop** that exploits the difference between estimated and expected features to drive the warping operators. Our overall approach is described in Figure 1.

#### 3.2. Estimators of visual motion features

The perceptual mechanisms by which humans look at, read, and understand images are well studied nowadays. Typical features such as chrominance, contrast and motion in the spectator field of view are well-known bottom-up key factors that influence audience attention. Attention is also driven by a number of top-down factors such as object semantics (faces draw strong attention), cultural background, and tasks to perform [39].

With this in mind, we propose **estimators** to measure *visual motion features*, that are computational characteristics designed to measure how well the motion of a character is perceived from an observer's viewpoint. Moreover, these features represent here a proxy for visual attention. As such, they provide a

view-dependent metric influenced by the character motion, the lighting in the scene and by potential occluders, and therefore provide a mean to control the amount of perceived motion in screen space.

For a given time frame  $t$ , we express the visual motion features as a time dependent vector  $s_t$ :

$$s_t = [s_1(t), \dots, s_V(t)]^T = s(\mathbf{m}(t), \mathbf{q}(t), \mathbf{\Omega}(t)) \quad (1)$$

where  $s_i(t)$  is a function of *a*) the character pose specified by  $\mathbf{m}(t)$ , *b*) the observer pose specified as a camera pose  $\mathbf{q}(t)$ , and *c*) the state of the environment  $\mathbf{\Omega}(t)$  that accounts for the rest of the scene, notably any lights and geometries that may affect the visibility of the virtual agent from the observer viewpoint. The function  $s(\mathbf{m}(t), \mathbf{q}(t), \mathbf{\Omega}(t))$  performs the scene and character rendering from the given camera, and computes the visual motion features.

*Visual motion features:*. In this work we consider the following features:

- *apparent static coverage* measures how much of a character's projected image is perceived in a frame (accounting for visibility and lighting);
- *apparent static extension* measures the horizontal occupancy (resp. vertical occupancy) as a ratio between the left-most and right-most pixels (resp. bottom-most and top-most) of the character on the screen width (resp. height), also accounting for visibility and lighting.
- *apparent motion coverage* measures how much of a character's motion from one frame to another is perceived in the image;

Likewise, designers are free to define and use additional visual motion features for their own application, at the condition that these can be represented as scalar values.

In practice, visual motion features are computed through hardware rendering and straightforward image analysis. At each time step, a frame is rendered from the observer's viewpoint and only the *perceived* pixels of the virtual agent are kept. A pixel is considered as *perceived* if it is not occluded, or if its luminance is under a given threshold (*e.g.* in a shaded or dark area). Visual motion features are estimated through pixel operations such as counting or comparing coordinates and distances in the image space.

*Semantic layers on body representations.* Sub-meshes of virtual characters are tagged so as to identify specific parts (face, arms, chest, legs, inside of hands). This enables us to arbitrarily activate or deactivate body parts according to the performed motion, *e.g.* to focus and render only the waving hand in a waving character case. In addition, rigid objects can be attached to the skeletal joints and visual motion features can be computed on them (*e.g.* an additional piece of clothing or holding a sheet of paper).

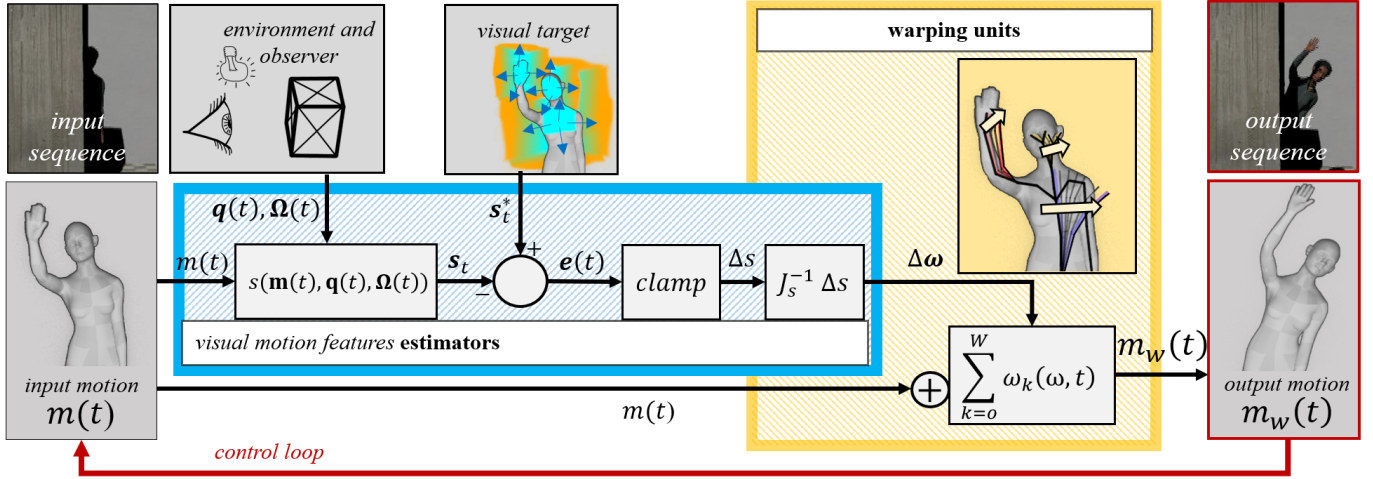


Fig. 1. Overview of our approach. From an input sequence of a character animation, we first estimate different visual motion features on the current pose, considering the environment, the observer's viewpoint, and a visual target (blue). Then, multiple plausible motion modifications are computed manipulating warping units (yellow), and the ones that minimise the visual error between the current state and the target are applied to the output motion. This process is repeated for the whole motion over a control loop (red).

### 3.3. Motion warping units

We first rely on a classical skeletal structure with joints using a tree of kinematic chains. An animation of the skeletal structure is defined as a set of keyframes along with an interpolation technique. We can thus define a function  $m(t)$  that computes the current pose of the character at time  $t$ , expressed as a vector of the degrees of freedoms of the characters joints denoted  $\theta_i$ :

$$\theta_t = [\theta_0, \dots, \theta_K]^T = m(t) \quad (2)$$

where  $K$  represents the number of degrees of freedom of the skeletal structure. Rather than regulating visual features by controlling simultaneously the whole vector  $\theta$  of joints of a character (which may create unexpected or unrealistic changes in the body poses), we propose to define specific groups of joints in the skeletal representation and define these as *motion warping units*. These groups are defined to provide a localised control on a character (e.g. only the spine, only the arms, only the head plus shoulders), which is a classical approach when designers need to locally warp motions without impacting the whole body (see Table 1). Furthermore, we design our motion warping units in a parameterised way that maintains the coupling between parameters of the kinematic chain by using a linear combination given a warping factor. This enables small warpings on the animations without losing the nature of the motion.

Our motion warping unit is therefore defined as a function  $\omega_k(\omega, t)$  that, given a scalar displacement value  $\omega$ , computes a sparse *pose offset vector* at time  $t$  affecting only the joint angles in the warping unit. For a given warping unit, the derived pose offset vector is added to the current pose  $m(t)$  to generate the warped animation. The index  $k$  represents the  $k$ -th warping unit, and  $\omega_k(\omega, t)$  yields a vector of size  $K$  with a zero-value for the degrees of freedom that are not offsetted. Our motion warping unit defines a pose offset vector in which each offset is a weighted linear combination of  $\omega$ :

$$\Delta\theta = \omega_k(\omega, t) = [w_0^k\omega, \dots, w_K^k\omega]^T \quad (3)$$

The scalar value  $w_i^k$  is a weighting constant specific to a given warping unit  $k$  and degree of freedom  $i$  of the kinematic chain and can be viewed as a stiffness coefficient, traditionally used when manipulating inverse kinematic chains. The linear combination with  $\omega$  ensures a coupling in the offset computation of animation parameters. The corresponding value  $\Delta\theta_i$  represents the  $i$ -th kinematic angle offset computed by the warping operator.

Each *motion warping unit* therefore computes a pose offset vector  $\Delta\theta_i$ , and all offset vectors are aggregated on the current character pose to create the warped motion (see Equation 4). The magnitude and direction of the computed offset vectors are driven by a vector  $[\omega_1, \dots, \omega_K]^T$  of warping unit parameters, the value of which is computed by the visual servoing task (see Section 3.4). The overall warped parameters  $m_w(t)$  of the animation at each time  $t$  are given by:

$$m_w(t) = m(t) + \sum_{k=0}^W \omega_k(\omega, t) \quad (4)$$

### 3.4. Driving warping units through visual motion features

Our objective is to compute the optimal set of warping unit parameters  $\omega$  at each time  $t$  of the animation from a given set of desired visual motion features  $s_t$ , through a **control loop**. We first express the relation  $s_t = f_i(\omega)$  where  $f_i$  computes the estimation of visual motion features. As a direct relationship between  $s_t$  and  $\omega$  exists, our goal is to solve this equation to obtain  $\omega$  from  $s_t$ . Due to the strong non-linearity of the relation between visual motion features and kinematic parameters, a classical approach is to study the problem in the velocity space.

Given this set of visual motion features  $s_t$  that depend both on a camera viewpoint at time  $t$  and a set of warping unit parameters  $\omega$ , the differential  $\dot{s}_t$  expresses how the variations in the visual features are related to the camera and the character animations.



*Introduction on visual servoing.* As defined, this problem is a specific case of a eye-to-hand visual servoing problem where a specified velocity in the image space of a fixed camera looking at a kinematic chain is used to drive its degrees of freedom [22]. This visual servoing relation is generally defined as:

$$\dot{s}_t = L_s V_n J_n(\theta) \dot{\theta} + \frac{\delta s}{\delta t}$$

where  $J_n(\theta)$  is the Jacobian of the kinematic chain,  $V_n$  the kinematic tensor transformation from the camera to the character,  $L_s$  the interaction matrix, and  $\frac{\delta s}{\delta t}$  describes the variations of  $s$  caused by a movement of the camera ( $n$  represents the number of degrees of freedom of the kinematic chain). This relation defines the correlation between the variation in visual motion features and the variations in degrees of freedom of the kinematic chain. We rely on this formulation to express our problem in terms of the warping unit parameters  $\omega$ .

$$\dot{s}_t = J_s \dot{\omega} + \frac{\delta s}{\delta t}$$

*Jacobian computation using finite differences.* Each element of the Jacobian matrix  $J_s$  encodes a partial derivative of *visual motion features* values ( $s$ ) over each *warping unit* feature ( $\omega$ ):

$$J_s = \left( \frac{\delta s_q}{\delta \omega_k} \right)_{q,k}, q \in [0..V], k \in [0..W] \quad (5)$$

where  $V$  is the total number of *visual motion features* and  $W$  is the total number of *warping units*. A forward evaluation enables us to compute a variation in visual motion features  $\Delta s$  from a variation in degrees of freedom  $\Delta \theta$  (for small enough variations):

$$\Delta s = J_s \Delta \theta \quad (6)$$

To solve the problem, we therefore reverse Equation 6 by approximating  $J^{-1}$  using a damped least square method, and we obtain the variation of degree of freedom, that we expressed as variation on  $\omega$  (linear combination of  $\theta$ , see Section 3.3).

$$\Delta \omega = J^{-1} \Delta s \quad (7)$$

The input vector  $\Delta s$  is classically computed as the difference between the expected features and the measured features  $e = s^* - s_t$  and capped with a maximum threshold:

$$\Delta s = \begin{cases} e & \text{if } \|e\| \leq DS_{max} \\ DS_{max} \frac{e}{\|e\|} & \text{otherwise} \end{cases} \quad (8)$$

In practice the computation of the Jacobian  $J_s$  at any time  $t$  requires to evaluate each *visual motion feature* for each  $2 * W$  variation of warping unit parameters. This is performed using finite differences, where variations in the agent motion need to be rendered to assess visual features.

We finally build the Jacobian matrix  $J_s$  as:

$$J_s = \left( \frac{s_q^+ - s_q^-}{2\omega_k} \right)_{q,k,t} \quad (9)$$

We then estimate the inverted Jacobian, and use it to extract the warping direction  $\Delta \omega$  (see Equation 7). An additional clamping is applied to the obtained vector to smooth the final motion modification. Finally, the new warped motion is computed using Equation 4. Implementation details and the setting of the expected feature  $s^*$ , for different case studies, are detailed in the following section followed by a user evaluation performed using Virtual Reality.

## 4. Results

We demonstrate our method over three case studies, with one virtual agent acting in front of one observer. We focus on upper-body nonverbal communication mainly using head, torso, arm and hand body parts, although the proposed method is suitable to control any body limb. Of environment conditions and related visual targets, we explore: i) the influence of changes in the observer's viewpoint, ii) the influence of occlusion or lighting conditions, and iii) the potential exaggeration of extraverted traits of a motion. Our accompanied video provides additional examples. Results and performances are discussed in this section, the method is more generally discussed in Section 6.

### 4.1. Implementation

We implemented our technique using *Unreal Engine*. The approach runs at interactive frame-rates (**>30fps on a computer equipped with an Intel Core i7-9850H CPU @ 2.60GHz, 32GB of RAM, Nvidia Quadro T2000**) and can be used in interactive and non-interactive contexts. **Estimators** (see Section 3.2) were implemented using shaders, while the visual motion feature vector was computed through multiple rendering passes. The **warping operators** were built above *Unreal* control rigs (see Section 3.3) which provide a direct access to the degrees of freedom of the skeletal structure of animated characters. Our virtual agent is based on a *Meta-humans* model. The baseline motions (unwarped) were either recorded using a motion capture system (Xsens suit) or taken from a public database (Mixamo). **During the execution, the control loop** (see Section 3.4) **generates**  $(2 * W) + 1$  copies of the virtual agent, at each frame, from the observer point of view: 1 copy is used as a reference (unwarped motion) for visual motion features evaluation and for the target definition, whilst the  $2W$  others are used to compute  $J_s$  by rendering warped motions, for each of the  $W$  warping units, in both warping directions.

To encode the different adapting behaviours, the designer first needs to decide the relevant *body parts* on which *visual motion features* are computed, then select the subset of these features to control. For each visual motion feature the designer can specify the magnitude and direction of its change over time, relative to the current value or to an absolute visual target. The magnitude affects mostly the responsiveness, and the direction defines the sign of the adaptation. Both values define the final visual target  $s^*$  (see Section 3.4). Consequently the designer needs to choose the set of active *warping units*, *i.e.* parts of the body which animations will be warped (see Table 1). We will describe and justify these choices in each example.

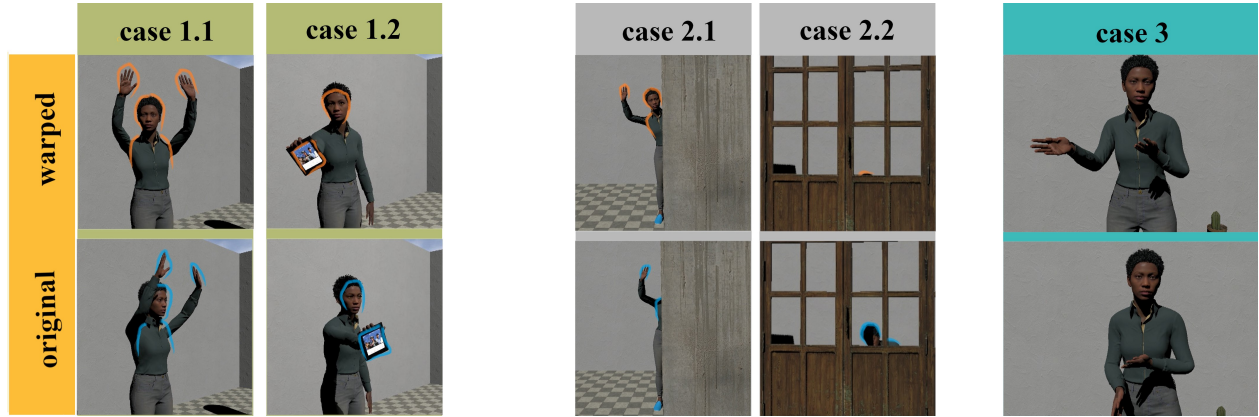


Fig. 2. Results from our three use-cases, displayed from the point of view of the observer. Character parts/objects on which the estimation of visual features is performed are highlighted. On the left (cases 1.1 and 1.2), we study the influence of viewpoint changes for a character with and without an object. The desired visual feature consisted in increasing the visual coverage. In the middle (cases 2.1 and 2.2), we study the influence of solid and sparse occluders. Visual coverage was also used as the regulating visual motion feature. On the right (case 3) we explore how extraverted effects can be created by applying upper body warping units, and regulating their apparent vertical and horizontal extension. **In the picture, the amplitude of the gesture of the warped character is wider with respect to the original non-warped one, making it feel more extraverted.**

Joint	Movement
Spine	Bend forward-backward
	Bend left-right
	Rot. vertical axis
Neck-head	Bend forward-backward
	Bend left-right
	Rot. vertical axis
Shoulders	Flexion-extension
	Abduction-adduction
	Medial-lateral rot.
Elbow	Flexion-extension
	Rotation on its axis
Wrist	Flexion-extension
	Ulnar-radial deviation
	Supination-pronation

Table 1. Defined pose warping units for upper-body motions. **The last three are independent for each arm.**

#### 4.2. Case studies

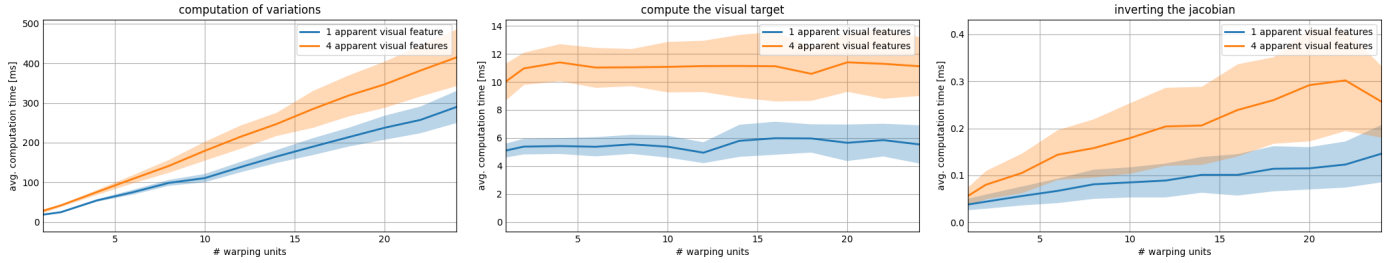
**Case 1 - viewpoint changes.** The purpose of this first study is to experiment the influence of viewpoint change on the animations to edit.

**Scenario 1.1:** In this scenario, the virtual agent aims at catching the attention of the observer while performing a two-hand waving motion. We recorded the baseline motion assuming that the agent was facing the observer. The objective is to modify this motion for a different camera angle, in a way that the agent better captures the observer's attention according to his/her position. We performed the visual evaluation with the face and the waving hands as body parts linked to the visual target. We selected *visual coverage* as the visual motion feature and specified a value to maximise frontal visual appearance. Also, when positioning the observer at different distances from the camera, we

used the control of the *apparent horizontal extension* to adapt the waving amplitude. The *warping units* related to the agent's spine, neck and arms were selected since they affect the waving motion of both hands.

**Scenario 1.2:** Here, the virtual agent shows to the observer an object held in its right hand. Again, our baseline animation was recorded with the observer facing the agent. The challenge is to adapt arm and hand motions to the observer's point of view in such a way that the object shown appears in the center of the observer's FoV. Regarding the relevant limbs related to the visual target evaluation, we selected the head and the additional external object (a tablet) held in the hand. Similarly to the previous scenario, we aimed for an increment of the *visual coverage* of the face and the tablet screen to maximise their visual appearance. The selected *warping units* were the ones related to the spine, neck and right arm of the agent since the motion was performed with this arm only.

**Informal analysis:** Scenarios 1.1 and 1.2 are illustrated in Figure 2 left. They were tested with different viewpoint parameters, namely, position, orientation, static/dynamic motions (see also our **companion video**). Results show that our approach successfully adjusts motions to changing viewpoints. Indeed, the first scenario demonstrates how the waving amplitude adjusts according to the observer's distance. In the second scenario we show how a unitary motion could be rapidly warped, to fit the duration of the action, with the same target of maintaining the visibility of a relevant object shown in the observer's FoV. We also show that our approach allows for the control of multiple limbs by generating subtle variations on a motion without affecting its original purpose. One could argue that similar results, especially Scenario 1.2, could be replicated using an inverse kinematics (IK) approach. This is only partly true, as our approach based on visual motion features also integrates environmental conditions such as lighting or scene layout (e.g. bring an object in front of someone, in light) with the exact same setting.



**Fig. 3.** Average computation time for three main parts of our technique, depending on the selected number of warping units. Left: time to compute the variations of visual feature values. Middle: time to compute the visual target in relation with the current visual feature values. Right: time to invert the Jacobian matrix. Results consider either one (blue) or four (orange) active apparent visual features.

**Case 2 - occlusion/visibility.** In this case study we aim at exploring through two scenarios the warping of agent limb motions to ensure proper visibility from the observer's viewpoint.

**Scenario 2.1:** In the first scenario, we recorded a one-hand waving motion, fully visible to a facing observer. The objective is to adapt this motion to improve its visibility by accounting for environment effects – occluding or partially occluding the animation. We considered as relevant limbs the ones that usually help capturing the observer's attention at a distance *i.e.* the waving hand, the face and the upper torso. For the same reason, the *visual coverage* was selected as the active *visual motion feature*. In this case, an increment of this feature for the hand, face and upper torso would improve the perceived visibility of the motion. The selected *warping units* influenced the spine, the right arm and hand.

**Scenario 2.2:** Here, our agent tries to hide from the observer; to achieve this, we captured a crouching motion as if someone was hiding behind an object, and used it for the original animation. The objective here was to adapt the motion to make the agent less visible from the observer's viewpoint by hiding behind occluders. A decrement on *visual coverage* (the selected *visual motion feature*) was selected as target. Finally, we also tried to simultaneously increment the perceived *visual coverage* of a specific limb while hiding the rest, *e.g.* maintaining eyes visible. We selected as *warping units* those related to the spine and the neck, and as relevant limbs all the upper body ones.

**Informal analysis:** Scenarios 2.1 and 2.2 are illustrated in Figure 2 center. These scenarios test different visibility parameters: static/dynamic occluders, static/dynamic lighting, different kinds of obstacles (see also our **companion video**). Results show that our method successfully adapts motions to visibility conditions and targets. Indeed, scenario 2.1 demonstrates how a waving motion initially recorded in a clear view situation facing a frontal observer can be adapted in other visibility conditions, by bending and adjusting the configuration of the arm towards a space where it was more visible. Scenario 2.2 shows how a crouching motion can be adapted to increase its hiding purpose, and we also show that our method enables combining this purpose with additional minor behaviours such as maintaining the top of the head visible. Such results demonstrate the advantage of our visual approach for these kinds of situations, by enhancing and adapting motions in different visibility conditions.

**Case 3 - expressivity.** Here, our aim is to experiment how visual features could be exploited to control the expressivity of a

motion with our approach. For the current example, we aim at influencing *extraversion*, one of the Big Five traits of personality [27]. This trait describes someone who typically captures the attention of an observer, is enthusiastic, energetic and sociable. In this scenario, the virtual agent is performing communication gestures as if it was talking. For the original animation we motion captured an actor conversing with the experimenter in a neutral way. The objective here is to adapt the arm and the hand motions to tune the trait *extraversion* of the agent, taking into account the observer's viewpoint. We selected *warping units* related to the head, elbow, shoulders, arms and hands, and the selected limbs were arms and hands. *Visual coverage*, *apparent vertical and horizontal extensions* were selected for the *visual motion features*. The visual target was then to increase/decrease both extensions to make the agent appear more/less extraverted. Indeed, Neff et al. [47] described several modifications to the character's gestures with a perceptual effect of creating an extraverted personality – by increasing spatial scale of the strokes, elbow rotations outwards (arm swivel) and increasing the shoulder raise. On the opposite, by decreasing the same characteristics, we define an introvert personality, additionally, for this case, we have controlled *Visual coverage* of the head to control the head posture.

**Informal analysis:** Scenario 3 is illustrated in Figure 2 right (see also our **companion video**). In this case study, we show that our method can modify gestures in similar ways than previous studies [47], which should also modify the perceived *extraversion* of the character.

#### 4.3. Performances

To objectively evaluate the computational load of our approach, we also present some information about the computational resources required for the application of our method.

The core computation runs on a dedicated thread at a frequency set between 3 Hz and 5 Hz, according to the selected parameters (*e.g.* warping units and visual features number). This thread is provided with the rendered scene and variations and outputs the warping parameters for the virtual character. At each frame, the separated rendering thread updates the posture of the character following the warping result.

Figure 3 shows the effects of the number of selected *warping units* (horizontal axis) and *apparent visual features* (blue and orange curves) on computation time. Values were captured on a machine equipped with Intel Core i7-9850H CPU



@ 2.60GHz, 32GB of RAM, Nvidia Quadro T2000. *Warping units* size has a stronger impact on performance for the computation of variations (order of magnitude  $10^2 ms$  in Figure 3 left), due to the need of generating and evaluating each variation of each warping unit. Nevertheless, the computation time is impacted linearly with the size of *warping units*. However, this heavy computation also further justifies the definition of *warping units* as a summary of the structure's degrees of freedoms, instead of computing variations on individual joints separately. On the other hand, the computation of the *visual feature's* level of the character, and the consequent target's definition (Figure 3 center), depends only on the number of the active *apparent visual features*. Finally, we show that the time to invert the Jacobian depends linearly on its dimensions  $V:W$  - number of *visual features* per number of *warping units* - and has minor impact on computation time (order of magnitude  $10^{-1} ms$  on Figure 3 right). In the presented implementation, the maximum possible size for the Jacobian matrix is 4:22, but none of the presented case required more than half of this dimensions.

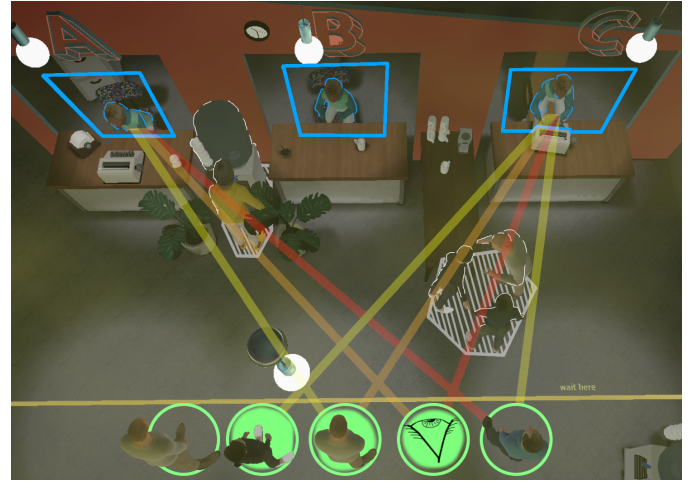
## 5. Evaluation

Since our main objective is to design reactive virtual humans who are able to interact and initiate an interaction with users, we designed an experiment in VR, where a virtual agent attempts to catch the user's attention by waving at him/her (similarly to case 1 and 2 from Section 4.2). In such a scenario, we aim at evaluating whether participants are able to detect if a waving motion is directed towards them or towards another agent. We also compare our technique to a standard approach based on orientating the root of the character. We hypothesize that our technique will outperform the simpler approach, showing higher a success rate to detect the target of the waving motion, as well as maintaining or increasing the naturalness of the interaction.

### 5.1. Procedure

Sixteen unpaid participants, volunteered to the experiment (3F, 13M; age: avg=25  $\pm$  3, min=20, max=30). They were all naive to the purpose of the experiment, had normal or corrected-to-normal vision, and gave written and informed consent. The study conformed to the declaration of Helsinki, and was approved by the local ethical committee (\*LEFT BLIND FOR REVIEW\*). Participants were first asked to read and fill up a consent form. They were equipped with a HTC-vive's headset, headphones and two controllers. The experiment was performed in a standing position and participants were able to take a break between each trial if needed. The evaluation takes place in a virtual office, as presented in Figure 4. The participant is spawned in a line of virtual customers, who are waiting to be called by one of the virtual office clerks, using a waving motion of the hand. In total there are three clerks, each one sitting behind a desk. In between the customers and the clerks there are additional agents and static objects that occlude the view of some of the clerks.

The participant stands in one of the three central positions of the customers group, and is exposed to one of the following



**Fig. 4.** The disposition of the agents in the virtual scene: starting from the bottom (green circles) the position of five customers, including the three (filled green circles) where the participant could be spawned. In the middle (highlighted in white) is the main area of visual occlusion, composed of objects and virtual agents. On the top: the three desks, each with a virtual employee (blue squares). Yellow, orange and red lines denote the degree of partial occlusion in the combined view direction between customers and clerks (yellow for low degree of occlusion, orange for medium, and red for high).

interactive situations: one of the employee waves in 3 possible directions: the participant (i), one of the other virtual customers to the left (ii) or to the right (iii) of the participant. The motion of the employee is animated using two techniques: a straightforward orientation of the whole body of the agent towards the target (SF\_adaptation), and our technique (visualWarping) parameterized with an increment in the *apparent visual coverage* of the head and the waving hand, using all the *warping units* affecting its two limbs. Additionally, for both techniques, we implemented the same gaze controller to orient the eyes of the agent towards the selected customer. In total, participants performed 2 training trials and 54 experimental trials presented in a randomized order (2 techniques  $\times$  3 directions  $\times$  9 repetitions). After each trial, participants were asked to: (i) answer the following question "According to you, to whom was the interaction directed towards?" with the possibility of selecting one of the customer's position, including their own, and (ii) to rate the level of realism of the interaction on a scale from 1 (low) and 9 (high).

At the end of the experiment, we also collected answers using a post-experiment questionnaire composed of questions related to demographics (gender, age and familiarity with VR), the strategy used ("What strategies did you use to identify to whom the interaction in the VR experience was directed to?") as well as self-reported free comments.

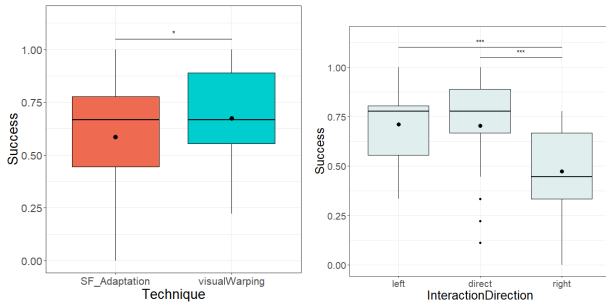
### 5.2. Analysis

For each participant and each condition, we computed the answer accuracy (i.e., the ratio of correct answers across repetitions), as well as the level of realism (the average level across repetitions). To evaluate the effect of the technique while considering the effect of the interaction direction of the waving motion, we conducted a two-way repeated measures Aligned Rank Transform (ART) Anova. When relevant, we performed Tukey

post hoc pairwise comparisons. Statistics were performed using R and the level of significance was set to 0.05. Results are presented using mean $\pm$ SD.

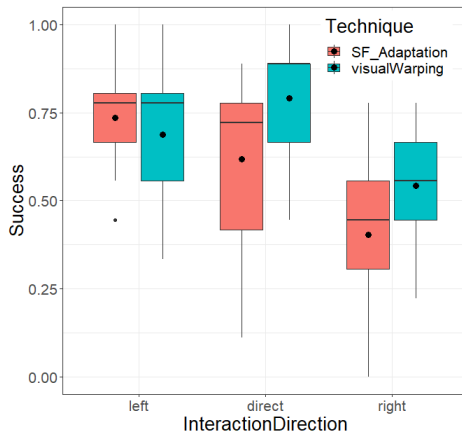
### 5.3. Results

Results showed a significant main effect of the technique on accuracy answers ( $F(1,15)=7.09$ ,  $p=0.017$ ,  $\eta_p^2=0.32$ ), with better accuracy with visualWarping ( $0.67\pm0.2$ ) than with SF adaptation technique ( $0.59\pm0.25$ ), as illustrated in Figure 5, left. There was also a main effect of interaction direction ( $F(2,30)=18.92$ ,  $p<0.001$ ,  $\eta_p^2=0.356$ ), where participants were less accurate when the virtual human was interacting to the right of the participants (Figure 5, right).



**Fig. 5. Main effects of the Technique and Interaction direction on participants' answer accuracy.**

There were no significant interaction effect ( $F(2,30)=2.19$ ,  $p=0.12$ ), but, as illustrated in Figure 6, there is a tendency for our method to be more efficient than SF Adaptation in direct and right relative interaction direction.

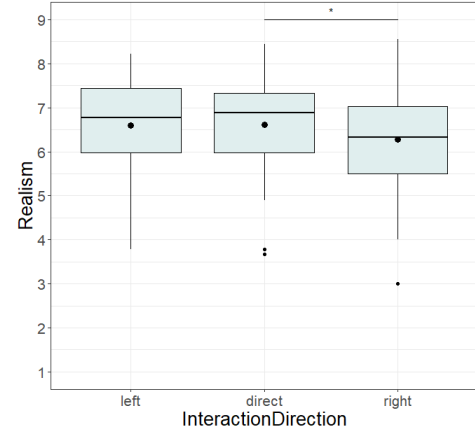


**Fig. 6. Participants' answers accuracy depending on the technique and interaction direction.**

Regarding the perceived level of realism, results showed only a main effect of the interaction direction ( $F(30,2)=3.97$ ,  $p=0.03$ ,  $\eta_p^2=0.21$ ), with a lower realism when the virtual human is interacting to the right than towards the participant (Cf. Figure 7).

### 5.4. Discussion

Overall, the presented results highlight the impact that vision-based editing techniques might have on human-agent interaction in Virtual Reality. Even though we have not detected



**Fig. 7. Main effect of interaction direction on the perceived realism.**

a relevant improvement in the perceived realism of the interaction, meaning that participants rated our edited interaction as realistic as the oriented one, we have observed a pertinent increase in the accuracy, thus participants could interpret easily the agent's aim when adapted with our technique. In apparent contrast with the current discussion, 100% of the participants mention eye's behaviour as one of the main and sometime the only (25%) feature used to identify the interaction's target. In fact this is not surprising, and even though eyes play a crucial role in user-agent interaction [7, 40], they are not always reliable, especially in situations when the proximity between the potential targets is narrow, or the eyes are not clearly visible (*e.g.* when the agent is far or not oriented towards the player). Hence, the results suggest that our technique can control body non-verbal communication, mentioned as secondary or primary feature in 75% of the answers, to enhance the interpretation of the waving orientation.

Moreover, it is important to discuss the disproportion we obtained between different relative directions of the target, for both techniques. We believe that this effect is caused by the design of the scene. Indeed, the right-most customers positions present, in general, a higher degree of occlusion relative to the clerks' desks, than the lefts ones (see Figure 4). Under this circumstance, we have reported a negative effect in realism and accuracy, for the relative right targets, which is partially mitigated in the accuracy by our technique.

## 6. Discussion and Conclusion

This paper proposes a motion warping technique that enables linking low-level motion variables with FoV-dependent visual motion features. Our results show that our approach is effective and applies to various cases, such as adjusting motions to changes in an observer's position, environment or lighting conditions. We also believe that this technique can be exploited to influence the expressions conveyed by animations (*e.g.* intro vs. extraversion) thereby helping designers to fine-tune the personalities of their characters or having virtual characters adapt their non-verbal communication towards observers (or avatars). Our proof of concept and the user experimentation in Virtual Reality validate the visual servoing scheme and yield a general and promising solution. While the method presents some analogies

with inverse kinematics methods, through-the-lens techniques, or line of action control, it offers higher levels of control than just positions and velocities of joints in the image space.

**Limitations and future works.** Currently, our method is limited in multiple ways. First, it requires a prior selection of visual motion features to be controlled, warping units to activate, or magnitude and direction of visual features. Methods to select relevant combinations of parameters would improve the practical usability of the approach. Inappropriately chosen parameters can generate unwanted results that are partially filtered out by joint limits and stiffness. Additional kinematic filters of motion editing (e.g. balancing the center of mass, remaining in the human motion manifold, preventing self-collisions) could help reduce the negative effects of parameter's tuning. In addition, we only explored a limited set of spatial visual features while operators could also perform time warping, and measure features over a sliding window rather than on a per-frame basis. Directly integrating computational saliency techniques [11] closer to visual attention mechanisms could also help to guide the warping of animations. Additional saliency biases could be added to account for top-down attention mechanisms specific to characters (e.g. focus of attention on head, eyes and hand movements) to build an attention-driven approach.

Second, the design of our warping units remains empirical. Existing work to automatically define rigging functions [35] could improve over our solution. In our case, we could explore means to automatically correlate low-level motion variables with the variations of visual motion features, with the difficulty that these relations depend on the motion performed, the desired goal of editing, and the observer's position.

At this stage, we also left apart the question of setting the *appropriate* levels of visual motion features editing. We partially address this question for specific use cases, i.e. the catching-attention scenario where the levels are set as if observed from an optimal angle, defined during motion capture. Still it remains challenging in other situations, in particular in the case of controlling motion expressivity. By which level a feature should be adapted to change the expression of motion? Which parameters or kinematics limitations enable reaching the desired editing levels? A data-driven approach would be relevant to address these questions. The difficulty is twofold: one is to gather the required amount of data to capture feature level variations with corresponding semantics, the other one is to deal with human variability in such behaviours.

In this work, we focus on upper-body motion, with the intention of control gestures that are salient for non-verbal interaction. However, we believe that the current paradigm, based on controlling high-level visual features, can be applied to an heterogeneous set of motion synthesis tasks: lower-body motion, i.e. steps generation, or action planner, i.e. which hand to use in the waving motion, etc. Such synthesis tasks would help improve the realism of the interaction.

In our examples, we explore the warp of a single character. Undoubtedly, it is relevant to discuss the case in which multiple characters are warped simultaneously. Considering a group of characters seen from a single point of view, our approach would adapt their configuration to optimize a joint set of ap-

parent visual features for that point of view. In terms of performances, this would require the evaluation of variations for each single character, but, as in Section 4.3, the total consumption would depend mostly on the total joint number of selected warping units. Rendering would not require additional computation. The dual case considers multiple point of views for a single character. Even though this approach is feasible, it would require the management of a compromise between possibly contradicting visual targets.

**Conclusion.** In conclusion, the technique we propose is a view-point dependent motion editing approach that exploits a number of visual features from an external observer's point of view to drive and warp animations. As a result, this can empower creative artists, but also autonomous characters with means to control the information they convey to an observer.

## Acknowledgments

Anonymised for review.

## References

- [1] Kevin Adistambha, Christian H. Ritz, and Ian S. Burnett. 2008. Motion classification using Dynamic Time Warping. In *2008 IEEE 10th Workshop on Multimedia Signal Processing*. IEEE, USA, 622–627. <https://doi.org/10.1109/MMSP.2008.4665151>
- [2] Rami Ali Al-Asqhar, Taku Komura, and Myung Geol Choi. 2013. Relationship Descriptors for Interactive Motion Adaptation. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Anaheim, California) (SCA '13). Association for Computing Machinery, New York, NY, USA, 45–53. <https://doi.org/10.1145/2485895.2485905>
- [3] Okan Arikan. 2002. Synthesizing constrained motions from examples. *ACM Trans. Graph.* 21, 3 (2002), 483–490.
- [4] Okan Arikan. 2006. Compression of Motion Capture Databases. *ACM Trans. Graph.* 25, 3 (jul 2006), 890–897. <https://doi.org/10.1145/1141911.1141971>
- [5] Okan Arikan and David A Forsyth. 2002. Interactive motion generation from examples. *ACM Transactions on Graphics (TOG)* 21, 3 (2002), 483–490.
- [6] Golam Ashraf and Kok Wong. 2000. Dynamic Time Warp Based Framespace Interpolation for Motion Editing.. In *Proceedings of the Graphics Interface 2000 Conference*. A K Peters/CRC Press, Wellesley, MA, USA, 45–52.
- [7] Steven A Beebe. 1974. Eye contact: A nonverbal determinant of speaker credibility. *Communication Education* 23, 1 (1974), 21–25.
- [8] Sarah-Jayne Blakemore and Jean Decety. 2001. From the perception of action to the understanding of intention. *Nature reviews neuroscience* 2, 8 (2001), 561–567.
- [9] Bobby Bodenheimer, Chuck Rose, Seth Rosenthal, and John Pella. 1997. The process of motion capture: Dealing with the data. In *Computer Animation and Simulation '97*. Springer Vienna, Vienna, 3–18.
- [10] Matthew Brand and Aaron Hertzmann. 2000. Style Machines. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*. ACM Press/Addison-Wesley Publishing Co., USA, 183–192. <https://doi.org/10.1145/344779.344865>
- [11] Neil DB Bruce, Calden Wloka, Nick Frosst, Shafin Rahman, and John K Tsotsos. 2015. On computational modeling of visual saliency: Examining what's right, and what's left. *Vision research* 116 (2015), 95–112.
- [12] Armin Bruderlin and Lance Williams. 1995. Motion Signal Processing. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '95)*. Association for Computing Machinery, New York, NY, USA, 97–104. <https://doi.org/10.1145/218380.218421>

- [13] Roser Cañigueral and Antonia F de C Hamilton. 2019. The role of eye gaze during natural social interactions in typical and autistic people. *Frontiers in Psychology* 10 (2019), 560.
- [14] Zhe Cao, Hang Gao, Karttikeya Mangalam, Qi-Zhi Cai, Minh Vo, and Jitendra Malik. 2020. Long-term human motion prediction with scene context. In *European Conference on Computer Vision*. Springer, 387–404.
- [15] François Chaumette and Seth Hutchinson. 2006. Visual servo control. I. Basic approaches. *IEEE Robotics & Automation Magazine* 13, 4 (2006), 82–90.
- [16] Byungkuk Choi, Roger Blanco i Ribera, John P Lewis, Yeongho Seol, Seokpyo Hong, Haegwang Eom, Sunjin Jung, and Junyong Noh. 2016. SketchMo: sketch-based motion editing for articulated characters. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–12.
- [17] Kwang-Jin Choi and Hyeon-Seok Ko. 2000. Online motion retargetting. *The Journal of Visualization and Computer Animation* 11, 5 (2000), 223–235.
- [18] Loïc Ciccone, Cengiz Öztireli, and Robert W Sumner. 2019. Tangent-space optimization for interactive animation control. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–10.
- [19] Beatrice de Gelder, AW De Borst, and R Watson. 2015. The perception of emotion in body expressions. *Wiley Interdisciplinary Reviews: Cognitive Science* 6, 2 (2015), 149–158.
- [20] Etienne Dombre and Wisama Khalil. 2013. *Robot manipulators: modeling, performance analysis and control*. John Wiley & Sons, Hoboken, New Jersey, USA.
- [21] Funda Durupinar, Mubbasir Kapadia, Susan Deutsch, Michael Neff, and Norman I Badler. 2016. Perform: Perceptual approach for adding ocean personality to human motion using laban movement analysis. *ACM Transactions on Graphics (TOG)* 36, 1 (2016), 1–16.
- [22] Bernard Espiau, François Chaumette, and Patrick Rives. 1992. A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation* 8, 3 (1992), 313–326.
- [23] Andrew W Feng, Yuyu Xu, and Ari Shapiro. 2012. An example-based motion synthesis technique for locomotion and object manipulation. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*. 95–102.
- [24] P. Giarion, R. Boulic, and D. Thalmann. 2004. PCA-based walking engine using motion capture data. In *Proceedings Computer Graphics International, 2004*. IEEE, USA, 292–298. <https://doi.org/10.1109/CGI.2004.1309224>
- [25] Michael Gleicher. 1998. Retargetting Motion to New Characters. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*. Association for Computing Machinery, New York, NY, USA, 33–42. <https://doi.org/10.1145/280814.280820>
- [26] Michael Gleicher and Andrew Witkin. 1992. Through-the-Lens Camera Control. *SIGGRAPH Comput. Graph.* 26, 2 (jul 1992), 331–340. <https://doi.org/10.1145/142920.134088>
- [27] Lewis R Goldberg. 1990. An alternative” description of personality”: the big-five factor structure. *Journal of personality and social psychology* 59, 6 (1990), 1216.
- [28] Martin Guay, Rémi Ronfard, Michael Gleicher, and Marie-Paule Cani. 2015. Space-time sketching of character animation. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–10.
- [29] Shihui Guo, Richard Southern, Jian Chang, David Greer, and Jian Jun Zhang. 2015. Adaptive motion synthesis for virtual characters: a survey. *The Visual Computer* 31, 5 (2015), 497–512.
- [30] Jinni Harrigan, Robert Rosenthal, Klaus R Scherer, and Klaus Scherer. 2008. *New handbook of methods in nonverbal behavior research*. Oxford University Press, Oxford, United Kingdom.
- [31] Robert A Hinde. 1972. *Non-verbal communication*. Cambridge University Press, Cambridge, England.
- [32] Edmond S. L. Ho, Taku Komura, and Chiew-Lan Tai. 2010. Spatial Relationship Preserving Character Motion Adaptation. *ACM Trans. Graph.* 29, 4, Article 33 (jul 2010), 8 pages. <https://doi.org/10.1145/1778765.1778770>
- [33] Daniel Holden, Ikhsanul Habibie, Ikuo Kusajima, and Taku Komura. 2017. Fast neural style transfer for motion data. *IEEE computer graphics and applications* 37, 4 (2017), 42–49.
- [34] Daniel Holden, Oussama Kanoun, Maksym Perepichka, and Tiberiu Popa. 2020. Learned motion matching. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 53–1.
- [35] Daniel Holden, Jun Saito, and Taku Komura. 2016. Learning Inverse Rig Mappings by Nonlinear Regression. *IEEE transactions on visualization and computer graphics* 23, 3 (2016), 1167–1178.
- [36] Yazhou Huang and Marcelo Kallmann. 2015. Planning motions and placements for virtual demonstrators. *IEEE transactions on visualization and computer graphics* 22, 5 (2015), 1568–1579.
- [37] Hongda Jiang, Marc Christie, Xi Wang, Libin Liu, Bin Wang, and Baoquan Chen. 2021. Camera keyframing with style and control. *ACM Transactions on Graphics (TOG)* 40, 6 (2021), 1–13.
- [38] Manmyung Kim, Kyunglyul Hyun, Jongmin Kim, and Jehee Lee. 2009. Synchronized multi-character motion editing. *ACM transactions on graphics (TOG)* 28, 3 (2009), 1–9.
- [39] Akisato Kimura, Ryo Yonetani, and Takatsugu Hirayama. 2013. Computational models of human visual attention and their implementations: A survey. *IEICE TRANSACTIONS on Information and Systems* 96, 3 (2013), 562–578.
- [40] Chris L Kleinke. 1986. Gaze and eye contact: a research review. *Psychological bulletin* 100, 1 (1986), 78.
- [41] Günther Knoblich and Natalie Sebanz. 2008. Evolving intentions for social interaction: from entrainment to joint action. *Philosophical Transactions of the Royal Society B: Biological Sciences* 363, 1499 (2008), 2021–2031.
- [42] Lucas Kovar and Michael Gleicher. 2003. Flexible Automatic Motion Blending with Registration Curves. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (San Diego, California) (SCA '03). Eurographics Association, Goslar, DEU, 214–224.
- [43] Lucas Kovar, Michael Gleicher, and Frédéric Pighin. 2002. Motion graphs. *ACM Trans. Graph.* 21, 3 (jul 2002), 473–482. <https://doi.org/10.1145/566654.566605>
- [44] Jehee Lee, Jinxiang Chai, Paul S. A. Reitsma, Jessica K. Hodgins, and Nancy S. Pollard. 2002. Interactive Control of Avatars Animated with Human Motion Data. *ACM Trans. Graph.* 21, 3 (jul 2002), 491–500. <https://doi.org/10.1145/566654.566607>
- [45] Yongjoon Lee, Kevin Wampler, Gilbert Bernstein, Jovan Popović, and Zoran Popović. 2010. Motion Fields for Interactive Character Locomotion. *ACM Trans. Graph.* 29, 6, Article 138 (dec 2010), 8 pages. <https://doi.org/10.1145/1882261.1866160>
- [46] Qianhui Men, Hubert PH Shum, Edmond SL Ho, and Howard Leung. 2022. GAN-based reactive motion synthesis with class-aware discriminators for human–human interaction. *Computers & Graphics* 102 (2022), 634–645.
- [47] Michael Neff, Yingying Wang, Rob Abbott, and Marilyn Walker. 2010. Evaluating the Effect of Gesture and Language on Personality Perception in Conversational Agents. In *Intelligent Virtual Agents*, Jan Allbeck, Norman Badler, Timothy Bickmore, Catherine Pelachaud, and Alla Safonova (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 222–235.
- [48] Sang Il Park, Hyun Joon Shin, Tae Hoon Kim, and Sung Yong Shin. 2004. On-line motion blending for real-time locomotion generation. *Computer Animation and Virtual Worlds* 15, 3-4 (2004), 125–138.
- [49] Tanmay Randhavane, Aniket Bera, Kyra Kapsaskis, Rahul Sheth, Kurt Gray, and Dinesh Manocha. 2019. EVA: Generating Emotional Behavior of Virtual Agents Using Expressive Features of Gait and Gaze. In *ACM Symposium on Applied Perception 2019* (Barcelona, Spain) (SAP '19). Association for Computing Machinery, New York, USA, Article 6, 10 pages. <https://doi.org/10.1145/3343036.3343129>
- [50] Claire L Roether, Lars Omlor, Andrea Christensen, and Martin A Giese. 2009. Critical features for the perception of emotion from gait. *Journal of vision* 9, 6 (2009), 15–15.
- [51] Charles Rose, Michael F Cohen, and Bobby Bodenheimer. 1998. Verbs and adverbs: Multidimensional motion interpolation. *IEEE Computer Graphics and Applications* 18, 5 (1998), 32–40.
- [52] Hubert PH Shum, Taku Komura, Masashi Shiraishi, and Shuntaro Yamazaki. 2008. Interaction patches for multi-character animation. *ACM Transactions on Graphics (TOG)* 27, 5 (2008), 1–8.
- [53] Adam Switonski, Henryk Josinski, and Konrad Wojciechowski. 2019. Dynamic time warping in classification and selection of motion capture data. *Multidimensional Systems and Signal Processing* 30, 3 (2019), 1437–1468.
- [54] H. Van Welbergen, B. J. H. Van Basten, A. Egges, Zs. M. Rutkay, and M. H. Overmars. 2010. Real Time Animation of Virtual Humans: A Trade-off Between Naturalness and Control. *Computer Graphics Forum* 29, 8 (2010), 2530–2554.



- 1 <https://doi.org/10.1111/j.1467-8659.2010.01822.x>  
2 arXiv:[https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-](https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8659.2010.01822.x)  
3 [8659.2010.01822.x](https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8659.2010.01822.x)
- 4 [55] Ruben Villegas, Jimei Yang, Duygu Ceylan, and Honglak Lee. 2018.  
5 Neural Kinematic Networks for Unsupervised Motion Retargetting. In  
6 *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*.  
7 IEEE, USA, 8639–8648. [https://doi.org/10.1109/CVPR.](https://doi.org/10.1109/CVPR.2018.00901)  
8 [2018.00901](https://doi.org/10.1109/CVPR.2018.00901)
- 9 [56] Andrew Witkin and Zoran Popovic. 1995. Motion Warping. In *Proceed-*  
10 *ings of the 22nd Annual Conference on Computer Graphics and Interac-*  
11 *tive Techniques (SIGGRAPH '95)*. Association for Computing Machin-  
12 *ery*, New York, NY, USA, 105–108. [https://doi.org/10.1145/](https://doi.org/10.1145/218380.218422)  
13 [218380.218422](https://doi.org/10.1145/218380.218422)
- 14 [57] Feng Zhou and Fernando De la Torre. 2012. Generalized time warping  
15 for multi-modal alignment of human motion. In *2012 IEEE Conference*  
16 *on Computer Vision and Pattern Recognition*. IEEE, USA, 1282–1289.  
17 <https://doi.org/10.1109/CVPR.2012.6247812>