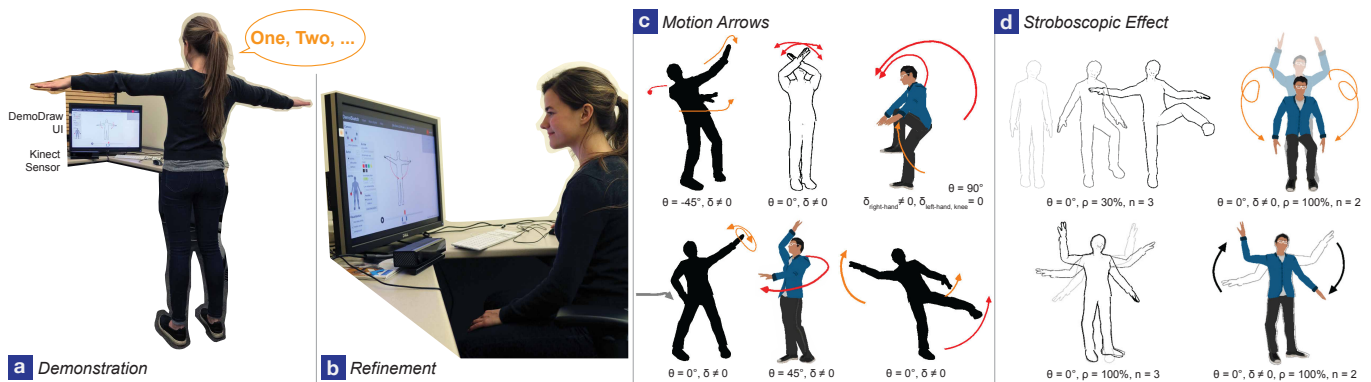


# Authoring Illustrations of Human Movements by Iterative Physical Demonstration

Pei-Yu (Peggy) Chi<sup>1</sup>, Daniel Vogel<sup>2</sup>, Mira Dontcheva<sup>3</sup>, Wilmot Li<sup>3</sup>, Björn Hartmann<sup>1</sup>

<sup>1</sup>Computer Science Division, UC Berkeley, <sup>2</sup>University of Waterloo, <sup>3</sup>Adobe Research  
 {peggychi,bjoern}@cs.berkeley.edu, dvogel@uwaterloo.ca, {mirad,wilmotli}@adobe.com



**Figure 1.** DemoDraw’s authoring interfaces and results: (a) multi-modal *Demonstration Interface* to capture motion, verify results, and re-perform portions if needed; (b) conventional *Refinement Interface* for refinement and exploring other visualization styles; (c-d) examples of illustration styles (annotated with camera viewing angle  $\theta$ , motion arrow offsets  $\delta$ , stroboscopic overlap ratio  $\rho$ , and numbers of intermediate frames  $n$ ).

## ABSTRACT

Illustrations of human movements are used to communicate ideas and convey instructions in many domains, but creating them is time-consuming and requires skill. We introduce DemoDraw, a multi-modal approach to generate these illustrations as the user physically demonstrates the movements. In a Demonstration Interface, DemoDraw segments speech and 3D joint motion into a sequence of motion segments, each characterized by a key pose and salient joint trajectories. Based on this sequence, a series of illustrations is automatically generated using a stylistically rendered 3D avatar annotated with arrows to convey movements. During demonstration, the user can navigate using speech and amend or re-perform motions if needed. Once a suitable sequence of steps has been created, a Refinement Interface enables fine control of visualization parameters. In a three-part evaluation, we validate the effectiveness of the generated illustrations and the usability of DemoDraw. Our results show 4 to 7-step illustrations can be created in 5 or 10 minutes on average.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

UIST 2016 October 16-19, 2016, Tokyo, Japan

© 2016 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-4189-9/16/10.

DOI: <http://dx.doi.org/10.1145/2984511.2984559>

## Author Keywords

tutorials; motion-capture; multi-modal interfaces; illustrations

## ACM Classification Keywords

H.5.2. User Interfaces: Training, help, and documentation

## INTRODUCTION

In sports, dance performance, and body gesture interfaces, movement instructions are often conveyed with drawings of the human body annotated with arrows or stroboscopic effects [19] (see Figure 2 for examples). These *illustrations of human movements* are also used within HCI to convey new user experiences in papers and storyboards [13]. When designed well, these illustrations can precisely depict the direction of motion while excluding unnecessary details such as clothing and backgrounds [19].

We found that both professionals and non-designers create these kinds of illustrations, but the methods they use are commonly time-consuming and not amenable to iteration and editing. The typical workflow is to prepare the physical scene, pose and photograph actors, and create annotated illustrations from the source photos. Even with the photos, producing effective depictions of the actors with integrated motion arrows and/or stroboscopic overlays takes considerable time and skill. Overall, the entire authoring process can take from 10 minutes up to several hours. Moreover, it can be difficult to identify the appropriate pose and viewpoint for the source photos before seeing the resulting illustrations. For example, one may

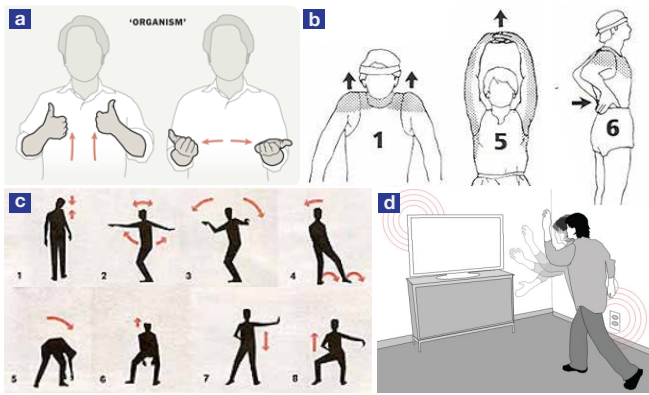


Figure 2. Examples of manually generated human movement illustrations: (a) for sign language [18]; (b) for weight training [4]; (c) for dance steps [unknown]; (d) for a gestural interface [17].

choose to exaggerate or change the orientation of a hand gesture after seeing the illustrated motion. Unfortunately, making such adjustments often requires starting over again with new source photos.

To address these challenges, we propose DemoDraw, a system that enables authors to rapidly create step-by-step motion illustrations through physical demonstration (see Figure 1). DemoDraw offers two key advantages over existing workflows. First, our system automatically renders characters and motion arrows based on demonstrations, which significantly reduces the amount of time and effort required to create an illustration. Second, DemoDraw helps users iteratively refine demonstrations to produce effective depictions. In our system, users can quickly add, replace, preview and modify demonstration takes.

Authoring proceeds in two modes: *Demonstration*, performed using body motions and voice commands; and *Refinement*, which uses a desktop interface. The user first physically demonstrates desired motions in front of a Kinect RGB-D sensor. As in current instructional practice, they simultaneously speak during important parts (e.g., teaching dance moves with “one, two, three, four”). The motions are then mapped to a 3D human avatar rendered as a black-and-white contour drawing, a common style identified in our survey of illustration practices. An algorithm analyzes speech and motion streams to segment motions into illustration figures with key frames. Salient joint movements are automatically identified and rendered as motion arrows overlaid on the stylized body drawing (Figure 4c). With this *Demonstration Interface*, segmented motions can be reviewed and re-recorded using speech commands. In addition, the annotation style and placement can be adjusted, camera angles moved, and alternate visualization styles explored in a mouse-driven GUI *Refinement Interface* (see Figure 5a). A three-part evaluation with 14 participants shows that DemoDraw’s illustrations are effective and amateur authors can use the Demonstration Interface and Refinement Interface to proficiently create illustrations of movements with various levels of complexity.

Our work includes the following specific contributions:

- An approach to generate human movement illustrations by direct physical demonstration and interactive rendering.
- Multi-modal interaction techniques to record, review, retake, and refine demonstration sequences.
- Methods to automatically analyze 3D motion data with speech to generate step-by-step annotated illustrations.

## RELATED WORK

Our work is related to research in demonstration-based authoring and motion visualization techniques.

### Demonstration-Based Authoring

User demonstrations have been harnessed to generate explanatory, educational or entertainment media in domains including software tutorials [9, 25], animation [8, 29], 3D modeling [42], or physical therapy [41]. For many of these systems, captured demonstrations are treated as fixed inputs that are then processed using fully or semi-automated techniques to produce a visualization. Work that falls into this category includes: generating step-by-step software tutorials from video or screen recordings with DocWizards [9], Grabler et al.’s system [25], and MixT [14], and automatically editing and annotating existing video tutorials with DemoCut [15]. This workflow is similar to graphics research that transforms existing artifacts into illustrations or animations. Examples include: using technical diagrams to generate exploded views [31], mechanical motion illustrations [35], or Augmented Reality 3D animations [36]; using short videos to generate storyboards [23]; creating assembly instructions by tracking 3D movements of blocks in DuploTrack [28]; and closely related to our work, using existing datasets of pre-recorded motion capture sequences to generate human motion visualizations with systems by Assa et al. [5, 6], Choi et al. [16], and Bouvier-Zappa et al. [11].

Animation is one domain where demonstration is often incorporated into the authoring workflow in a more interactive manner. For example, GENESYS [7], one of the earliest computer animation systems, allows users to perform motion trajectories and the timing of specific events with sketching and tapping interactions. Performance-based animation authoring remains a common approach, and recent work shows how physical props can be incorporated to support layered multi-take performances [20, 27] and puppetry [8, 29].

While the primary goal of performance-based animation systems is to accurately track and re-target prop motions to virtual characters, DemoDraw focuses on the mapping from recorded body movement demonstrations to static illustrations conveying those motions. Some previous systems have also mapped body movement to static media: BodyAvatar [42] treats the body as a proxy and reference frame for “first-person” body gestures to shape a 3D avatar model and a Manga comic maker [32] maps the body pose directly into a comic panel. Systems using interactive guidance for teaching body motions are essentially the inverse of DemoDraw. Examples include YouMove [3] that teaches moves like dance and yoga, and Physio@Home [38] that guides therapeutic exercises.

## Motion Visualization

Several of the systems above focus on developing automated algorithms to visualize various dynamic behaviors, such as mechanical motion [31, 35, 37], motion in film [23], molecular flexibility [12], and human movements [5, 11, 16]. Much of this work is inspired by formalizing techniques and principles for hand-crafted illustrations [2]. Bouvier-Zappa et al.'s [11] automatic approach visualizes large collections of pre-recorded motion capture sequences. We support many of the same visualization techniques, including motion arrows, overlaid ghosted views, and sequences of poses, but we introduce an interactive approach for authors to create illustrations for particular motions to share with others. Since such demonstrations often involve mistakes and repeated takes of the motion, DemoDraw supports interactions to help authors review and retake portions of their demonstrations. Moreover, the interactive nature of DemoDraw enables more fine-grained controls for adjusting visualization parameters and compensating for idiosyncratic characteristics of automated algorithms.

## MOTION ILLUSTRATION PRINCIPLES AND METHODS

To understand motion illustration design and production, we surveyed related literature, studied collected examples, and interviewed individuals who create such illustrations.

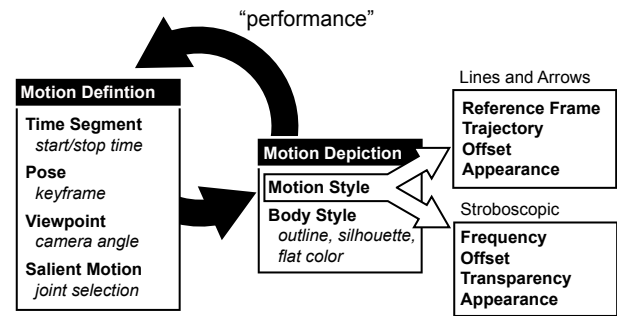
### Design Principles

Cutting [19] argues that superimposing vector-like lines, often called “actions lines”, on an image satisfies four important criteria: it evokes a feeling of motion, the object undergoing motion is clearly represented without deformation, the direction of motion is clear, and the magnitude of motion is conveyed with reasonable precision. To complement this metaphoric representation, Cutting also argues for the more literal method of multiple stroboscopic images, which satisfies all criteria except clear motion direction. McCloud [33] provides further arguments and examples for using these methods in the field of comic illustration, and notes communication benefits when they are combined.

To examine how professional illustrators use motion lines and stroboscopic images, we gathered examples from sources like user manuals, gesture-based games, safety guides, illustration compendia (e.g., [34]) and how-to books (e.g., [26]). We found Cutting’s notion of vector-like lines are almost always rendered with an arrowhead in a variety of styles (heads, weights, colors) with strokes typically two-dimensional, smooth, and offset to avoid occluding the object. Stroboscopic images can be overlapping or spatially distributed, and change in transparency or shading to convey time. The most common style for depicting the object undergoing motion is a simplified black-and-white contour drawing, but filled silhouettes and flat-shaded colour can also be found – using full color photographic detail is rare. By carefully removing extraneous details, such techniques help readers focus on only the salient motion information.

### Interviews: Methods Used In the HCI Community

Conveying movement for interaction is common in HCI publications. We found 100 motion illustrations in 58 recent papers. To understand current creation methods, we conducted video



**Figure 3. Canonical authoring workflow consisting of a Motion Definition task then a Motion Depiction task. Design decisions associated with a task are shown in bold with design parameters in italics.**

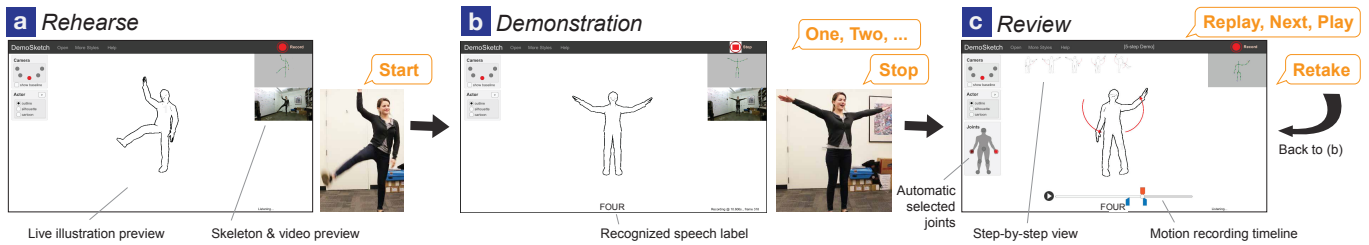
interviews with six Human-Computer Interaction researchers with experience creating motion illustrations.

**Findings.** All interviewees used a similar methodology to create motion illustrations: they took still photographs of people performing actions, traced outlines using Adobe Photoshop (4/6) or Illustrator (2/6), then added graphic annotations to convey motion. All mentioned that it was time-consuming to set up scenes and poses, take and trace photos, then add details like arrow placement while maintaining a consistent style. Typical creation times were estimated between 10 minutes to a few hours. They also noted how difficult it was to make adjustments: changing the pose or viewpoint essentially meant starting over again with new source photos and re-tracing. Yet, identifying the best pose and viewpoint ahead of time is difficult and it often took several iterations to yield an illustration suitable for publication.

### Design Space Goals and Workflow

Based on the observations above, we derive a canonical workflow to motivate our system’s central design goal. Authors face two primary illustration tasks (Figure 3): *defining the motion* for portraying movements like the view of the body and salient moving joints; and *exploring a style of motion depiction* by choosing styles like lines-and-arrows or stroboscopic, then adjusting related style parameters. These tasks and the underlying design parameters are highly interdependent, so authoring motion illustrations is necessarily an iterative process. This means that changes to one task parameter often leads to re-evaluating and changing the other. The problem with current methods, is that movements are mostly “performed” using a time-consuming process of taking photos and manually tracing them. Therefore, the central design goal of our system is to make motion definition low effort and iterative via interactive demonstrations.

Designing a system to capture interactive demonstrations of *any* body movement also poses an input challenge. Since body movements form the demonstration itself, also issuing application commands with a body gesture introduces ambiguity. Using a hand held device, touch screen, or any conventional input is not ideal since performing requires open space and full freedom of movement. For these reasons, we use a multi-modal voice and gesture interaction style traced back to Bolt’s



**Figure 4. DemoDraw authoring UI: Using the Demonstration Interface, an author sees an avatar following her real-time movement (a). During recording (initiated by voice command “Start”), real-time feedback shows the speech labels (b). Once a recording is completed by voice command “Stop”, the motion visualization and a timeline are immediately available (c) for the author to review, and a step-by-step overview will be generated.**

Put-That-There [10]. Like Bolt, we use voice for commands like “start” and “stop” with body movements providing command parameters in the form of the recorded demonstration, and for setting parameter context with utterances like “one, two, three, four” to label step-by-step segments.

### DEMODRAW

DemoDraw is designed for non-experts who cannot effectively or efficiently create concise motion illustrations motions using existing tools. To provide an overview of how the system works, we present a scenario in which a motion illustration author, Marie, creates instructions for an 8-step dance tutorial.

In her living room, Marie begins using DemoDraw with the Demonstration Interface shown on her television by standing in front of a Kinect. In the center of the display, an avatar follows her movements in real-time (Figure 4a). This avatar is shown as an “outline” figure, but she could always change to different rendering effects like “silhouette” or “cartoon,” or select a different 3D human model later using our Refinement Interface (Figure 5a).

**Recording.** Marie starts recording her physical demonstration with the voice command “Start.” After a 3-second countdown, DemoDraw captures the position, orientation, and depth distance of her body (using Kinect’s simplified 25 body joint model). While demonstrating dance moves, Marie verbally indicates the count of each step with “one, two, three, and four,” just like she does when teaching a dance. The specific utterance is not constrained, Marie could use words like “right, left, shake, and clap.” A speech recognition engine captures these labels with timestamps and displays them in the interface (Figure 4b). Marie finishes recording by saying “Stop”.

**Reviewing and Re-Recording.** After recording, DemoDraw automatically segments the motion around the speech labels and identifies salient joints. An illustration of the first step of Marie’s demonstration is rendered with motion arrows, showing the path of the most salient joints. Figure 4c presents an example illustration that shows how her right hand waves from bottom to the top, and the left on the opposite direction. She also notices three panels emerged: A timeline below shows the start, end, and key frame points used to generate the current illustration, a side panel shows the visualized joints; an step-by-step overview of step snapshots is created and added to a motion sequence list. Marie can navigate to other illustrated steps by either saying “Next” or “Back”, or repeating one of the words she said during recording (like “three”) to skip to

that corresponding step. To play an animation showing her continuous motion, she can say “Play” to play the current step only, or “Replay” to play the entire motion recording with each step visualization highlighted.

Once Marie reviews the steps, she realizes she should have exaggerated the hand motion in step 4. By saying “Retake Four,” Marie can re-record a partial sequence of movements including that step (e.g., redoing and saying “Four” and “Five”). When she ends the re-recording with “Stop”, the old illustration for that step is replaced with a new one (step four in this example) generated using the new motion recording.

**Motion Depiction Adjustments.** Once Marie is satisfied with her demonstration, she walks out of the capture area to her desktop computer. The system automatically switches to the Refinement Interface by revealing post-processing panels in a standard graphical user interface (Figure 5a). Using this interface, Marie can adjust several design parameters: the arrow appearance can be refined, including line width, arrowhead size, and color; the arrow offset can be adjusted with direct manipulation dragging; the camera viewpoint can be adjusted by orbiting the camera to a side or three-quarter view; the joints used for motion paths can be added or removed using a panel; and the smoothed motion trajectory can be toggled on and off. She could also select a different key pose and adjust the start and end times of a motion segment by dragging the markers on the timeline. In addition, Marie could explore other illustration styles like stroboscopic rendering by selecting numbers of intermediate frames and how they render in one diagram (Figure 5b). These results can be exported to image files containing the final motion illustrations.

### SYSTEM COMPONENTS AND IMPLEMENTATION

DemoDraw has four main components (Figure 6): a *motion capture* engine to record joint data from the author’s demonstration and apply it to a 3D avatar; a *speech recognition* engine to process speech input for commands and motion labels; a *motion analysis* algorithm to partition recorded motion and identify salient joint movements for each illustration segment; and an *illustration rendering* engine to visualize the avatar and motion segments with different effects. These components combine into an interactive and iterative system pipeline to translate demonstrations into motion diagrams. A notable technical contribution is our motion segmentation algorithm combining speech labels and joint motion streams. DemoDraw is implemented using C# in Unity 5. It runs interactively

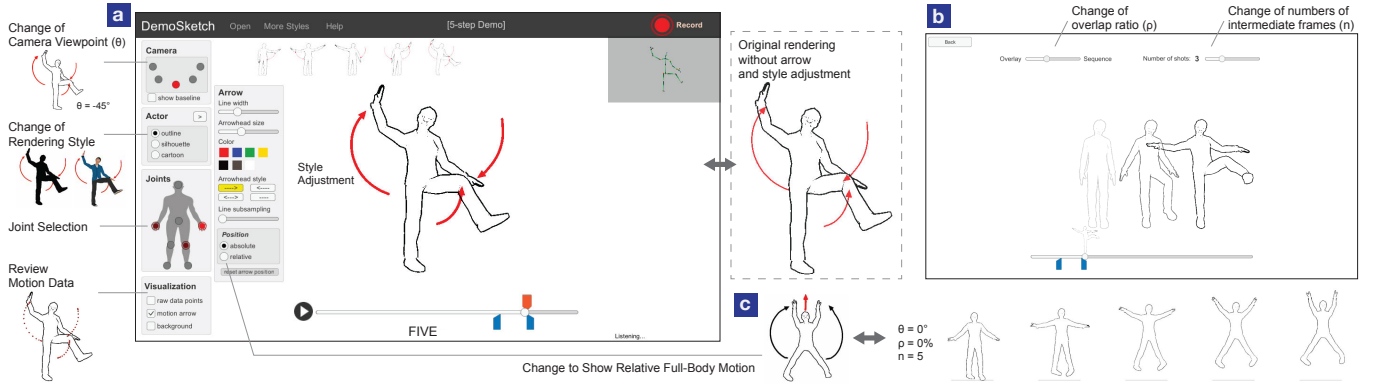


Figure 5. Using DemoDraw’s Refinement Interface, the author can refine the visuals (a) and explore more illustration effects (b, c).

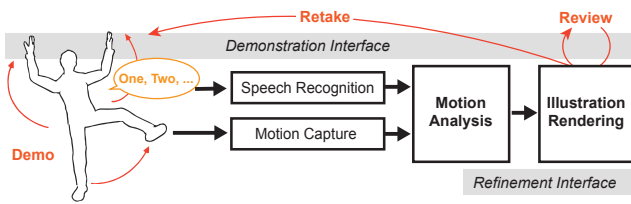


Figure 6. DemoDraw system components and pipeline.

on a Macbook Pro with Windows Bootcamp (2.5 GHz Intel Core i7 processor and 16 GB memory). Below we describe the design and implementation of each component.

### Motion Capture

In support of our design goal to enable low-effort iteration within tasks, the motion capture component provides real-time feedback during demonstrations so authors can monitor their performance accordingly. We capture position and joint angles of a simplified 25-joint skeleton using a Kinect2 sensor and the Kinect SDK 2.0. The real-time joint data is applied to a generic 3D human model (an “avatar”) using forward kinematics enabled by a modified Unity asset<sup>1</sup>.

### Speech Recognition

Speech is used when recording a demonstration to label motions (e.g., “one, two, ...”) and for recording and navigation commands (e.g. “Start, Stop, Retake” or “Replay, Next, Play”) – see Figure 4 for the speech commands that DemoDraw supports. We recognize both types of speech using the Microsoft speech recognition library<sup>2</sup> to process audio captured by the Kinect microphone array. During recording, the start time, duration, and confidence of each motion label are logged for use in the motion analysis algorithm.

### Motion Analysis

Our motion analysis algorithm translates a multi-part demonstration recording into a sequence of labeled time segments, each with one or more salient joint motions and a keyframe of joint positions for a representative body pose (see Figure 7

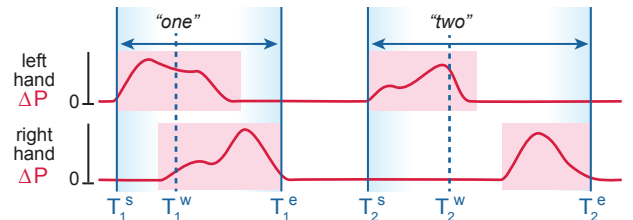


Figure 7. Illustration of motion analysis algorithm (two joints shown due to space): significant moving periods of joint movements (pink) are mapped to speech labels to define motion segments (blue). Note the right hand period is mapped to “two” because it begins shortly after the left hand period.

for an illustration of the approach). Formally, given a set of  $n$  speech labels  $\{w_1, w_2, \dots, w_n\}$  that end at latency-corrected times  $\{T_1^w, T_2^w, \dots, T_n^w\}$ , our algorithm associates each speech label  $w_i$  with a *motion segment*, of which the start and end time are denoted as  $[T_i^s, T_i^e]$  where  $T_i^s \leq T_i^w \leq T_i^e$ . Each motion segment includes a set of  $k$  salient joints  $\{j_1^i, \dots, j_k^i\}$  and keyframe time  $T_i^{key}$  between  $[T_i^s, T_i^e]$ . It is then sent to the Illustration Rendering engine to create a motion illustration in a multi-part sequence.

Human motion segmentation and activity understanding has been well studied in computer vision and graphics [1]. We adopted a spacetime approach to identify salient motion sequences in 3D space. However, in our scenario such as dancing, movements may not necessarily encode a semantic meaning for automatic recognition, such as “walking” or “throwing (a ball)” in previous research. Therefore, our approach combines the user’s speech labels, similar to a scene segmentation method used in DemoCut [15]. We make two assumptions about the synchronized data streams of speech labels and joint movements: 1) authors make short pauses between motions to be grouped, i.e.,  $T_i^e < T_{i+1}^s$ , and 2) the speech label utterances overlap or closely occur with at least one joint motion. These assumptions are practical since authors often pause for a moment to prepare for demonstrating the next movement in a step-by-step sequence.

**Motion Segmentation.** To determine a motion segment of  $[T_i^s, T_i^e]$  for each speech label  $w_i$  that ends at  $T_i^w$ , we begin by

<sup>1</sup><https://www.assetstore.unity3d.com/en/#!/content/18708>

<sup>2</sup><https://msdn.microsoft.com/en-us/library/hh361572>

identifying all *moving periods* of significant joint movements (pink rectangles in Figure 7) for 8 joints  $J$ : the 5 end-effectors (head, hands, feet), 2 knees, and the body root. To filter jittery movements, joints are considered moving if smoothed inter-frame differences in absolute Euclidean distance are greater than a threshold. Specifically, for each joint  $j \in J$  of a frame  $r$  at time  $t$ , the average difference in position between two adjacent frames  $\Delta P = |P^r - P^{r-1}|$  is computed over the subsequent half second (15 frames). If this moving average is greater than  $0.05m/s$ , then joint  $j$  of a frame is labeled as “moving”, marked as  $m_j^r$ . This is repeated on all frames and all joints. Next, of the entire motion recording for joint  $j$ , we combine all the consecutive  $\{m_j^r, m_j^{r+1}, \dots\}$  into a joint moving period  $M_j$ .

Once a list of moving periods  $\{M_j^1, M_j^2, \dots\}$  for joint  $j$  is determined, we begin labeling each  $M_j^m$  at  $[T_m^s, T_m^e]$  to map to a speech label  $w_i$  at time  $T_i^w$  where  $T_m^s \leq T_i^w \leq T_m^e$ . In other words, the speech utterance occurs during or near to a joint movement (illustrated as dashed lines crossing pink rectangles in Figure 7). After all moving periods are mapped to speech labels for all major joints in  $J$ , the start and end time  $[T_i^s, T_i^e]$  of the motion segment for label  $w_i$  are set to the minimum start time and maximum end time across all mapped joint movement periods.

**Joint Salience Identification.** The salient joints  $\{j_i^1, \dots, j_i^k\}$  are defined by the set of all joints that were mapped based on significant moving periods.

**Key Pose Selection.** A key pose is used to represent a motion segment in an illustration. Based on our informal experiment, it is often the end state of movements as motion arrows are pointed toward this end goal (see the Figure 6 for example). Therefore, we set a key pose at the end of a motion segment, i.e.,  $T_i^{key} = T_i^e$ .

**Motion Retake.** When retaking a partial demonstration with one or more speech labels  $\{w_i', w_{i+1}', \dots\}$ , the full motion analysis algorithm is run on the new recording. New motion segments then replace the original segments by mapping  $w_i'$  with  $w_i$ .

### Illustration Rendering

The Illustration Rendering engine generates a motion illustration for each motion segment of speech label  $w_i$  (bounded by  $[T_i^s, T_i^e]$ ). There are two related rendering tasks: the body pose and the motion depiction style.

**Body Pose.** The body pose is determined by all joint positions at keyframe time  $T_i^{key}$ . We use standard Non-Photorealistic Rendering (NPR) [24] techniques to render the 3D human model in a stylized manner that abstracts away distracting details. Specifically, we support contour-only, filled silhouette, and flat-shaded rendering styles (see Figure 5a left for examples).

**Line and Arrow Depiction Style.** Based on Cutting’s criteria [19] and our survey of motion illustrations, we use lines with arrowheads as the default depiction style for visualizing joint movements. This style is rendered as follows: For each

salient joint of a motion segment, the absolute joint positions in world space over the period  $[T_i^s, T_i^e - \epsilon]$  are used to construct a 3D poly-line using Catmull-Rom interpolation. Rather than visualizing the entire path, we set  $\epsilon$  to be 0.5 seconds to visually point the arrow toward the key pose at  $T_i^e$ . Two 3D cones are positioned collinear with the last two polyline positions to form arrowheads for both the beginning and the end of a line. Although the poly-line is 3D, it is shaded to appear 2D. All arrows are colored red by default to contrast with the avatar, a common technique for layering information [40].

For some motions, visualizing absolute joint positions might not be suitable. For example, for a two-foot jump with a two-hand waving motion (see Figure 5c), our algorithm will mark all major joints as salient and generate multiple arrows showing the jump movement, but fail to convey the hand waving. Authors can choose to visualize joint motions *relative* to the spine instead, triggering the same motion analysis algorithm described above to be re-run using relative motion. In this way, the same movements would be shown more concisely with a single up arrow (for the overall jump direction) and two curve arrows (for the hand movements).

**Other Adjustments.** Authors can review the results using the Demonstration Interface or Refinement Interface. With the latter, line weight, arrowhead sizes, and color can be adjusted and re-rendered in real-time using graphical widgets (see Figure 5a). Arrows can also be re-positioned to increase the offset ( $\delta$ ) by direct manipulation dragging. Considering some movements cannot be easily seen from the default front camera viewpoint (such as those parallel to the XZ plane, see Figure 1c top-right), our UI enables the selection of four other camera angles ( $\theta$ ), including three-quarter front views ( $45^\circ$  and  $-45^\circ$ ) and profile views ( $90^\circ$  and  $-90^\circ$ ), all at the eye level. These discrete choices simplify control, but of course it would be possible to select any viewing angle given the 3D avatar and joint information. By default, 8 main joints are analyzed and illustrated, but any of the 25 body joints can be explicitly selected for illustration using the interface.

**Stroboscopic Depiction Style.** Cutting [19] noted stroboscopic effects are also effective, and we found examples of illustrations with a sequence of overlaid semi-transparent body poses in our survey. Therefore, authors can select a stroboscopic depiction style in the Refinement Interface (see Figure 5b). The style is rendered by compositing multiple semi-transparent renderings of intermediate body poses between  $T_i^s$  to  $T_i^e$  behind a rendering of the representative pose at keyframe time  $T_i^{key}$ . Authors can adjust the number of intermediate poses  $n$  (the default is 3 poses) and the horizontal overlap ratio  $\rho$  between intermediate pose renderings can be adjusted to stack them up ( $\rho = 100\%$ ) or spread them out ( $\rho = 0$  is the default).

### Results

The DemoDraw pipeline is capable of generating expressive and clear motion illustrations. In Figure 1c, motion arrows show the upper body motion (top left), hand waving back and forth (top middle), and hand circular motion (bottom right). Whole body motions can also be visualized (bottom left), and can be especially helpful when motions are best viewed from a

different angle, such as the side view (top right). In Figure 1d, stroboscopic effect depicts the transition from the start pose to the end pose, which can be rendered as a sequence (top left) or in one combined pose (bottom left). A combination of this effect with motion arrows creates a compact, integrated illustration (top and bottom right).

## USER EVALUATION

We evaluated the capability and usability of DemoDraw in three lab-based studies. The first study with 10 participants tested the effectiveness of illustrations generated by DemoDraw (*How well do users understand static motion illustrations?*). The second study with the same participants evaluated the Demonstration Interface for recording motion demonstrations (*Can users generate step-by-step illustrations with our system?*). The third study with 4 different participants evaluated the Refinement Interface for editing a pre-captured recording (*Can users refine illustrations with our system?*). Recall that our survey found current methods (using software like Adobe Illustrator) are time intensive, require design expertise, and make iteration difficult. For these reasons, we did not include a baseline in our evaluations.

### Study 1: Illustration Effectiveness

We hypothesized that learners can understand and re-perform motions after reviewing step-by-step illustrations generated by DemoDraw. To validate, we recruited 10 participants (5 females), aged 18 to 33 years ( $M=24.3$ ), from a university and an IT company. Six participants had previously created illustrations (from 5 to 50 diagrams, typically using Adobe Illustrator), but none involved body motion. We first showed the illustrations in Figure 2 to introduce the context, then we presented two sets of printed diagrams generated by our the experimenter using our system. There were 16 highlighted joint movements in 8 steps in total (see Figure A.1). For each set, participants interpret the illustrations and performed the movements in front of a video camera.

**Measures and Results.** We coded each joint movement using the video recordings as follows: (M1) the user moved the correct joint; (M2) the start and end positions were approximately correct; and (M3) the movement was performed correctly (e.g., moving hand straight). Out of 160 re-performed motions across all 10 participants, 85% motions were completely correct (i.e., M1, M2, M3 all correct). On average, each participant performed 87.2% of the motions correctly ( $sd=7\%$ ). No users intentionally moved non-annotated joints. Table 1 shows 6 motions that resulted in errors. These motions fall into two classes: temporal sequencing of multiple joints moving in a single step; and cyclic actions (e.g., waving). Some participants could read and re-perform these more complex motions, but there may be limits to what motion diagrams can convey.

### Study 2: Demonstration Interface

We hypothesized that amateurs can efficiently create motion illustrations using DemoDraw’s multi-modal Demonstration Interface. Immediately after Study 1, the same 10 participants completed this study (total time was 45 to 60 minutes).

Study 2 began with training where participants were introduced to DemoDraw and shown how to create the second set

|               | Error type and rate | Intended motion                                  | Performed motion                                    | Participant Explanation  |
|---------------|---------------------|--|---|--|
| Set I Step 2  | M3 (2/10)           | Waving hands back and forth                      | Waving one way (crossing hands)                     | Didn't catch it  |
| Set I Step 3  | M3 (4/10)           | Hands circling out and in                        | Hands circling in and then out (opposite direction) | Didn't catch it  |
|               | M1 (7/10)           | Squatting  | N/A (missed)  | (6/7) focused on hand motions and did not see the down arrow; (1/7) noted the arrow but thought it referred to the hands |
| Set II Step 1 | M2 (5/10)           | Moving right hand from lower left to upper right | Moving right hand from lower right to upper right   | (4/5) Didn't notice the start position; (1/5) assumed the starting pose is a stand position                              |

Table 1. Incorrect movements performed by participants in Study 1.

of illustrations from Study 1. Participants then recorded the same motions and reviewed the results (training was 5 to 10 minutes). Then, four tasks were completed in sequence:

1. The experimenter demonstrated 4 moves for a gestural interface with their right hand: waving, circling, a reversed V shape, and right swipe (see Figure A.2-1). Participants were asked to record these motions and review the captured results. Once satisfied, they rated the generated illustrations.
2. Similar to task 1, but with 8 dance moves (see Figure A.2-2).
3. The experimenter introduced the retake operation and asked participants to choose one step from task 2 to revise. Participants re-performed the motion and reviewed.
4. Participants were asked to perform any 4 to 8 moves they could imagine and retake them until they were satisfied with the results (within a time limit of 5 minutes).

The system was displayed on a 30-inch monitor (with mouse and keyboard) and the Kinect sensor was placed 3-feet above the floor, capturing 8×8-feet of clean space.

**Measures.** In tasks 1 and 2, participants rated each step along five dimensions: (Q1) “The visualization accurately captured/described my motion”, (Q2) “The visualization shows all the important joints of movement”, (Q3) “It shows at least one extraneous joint”, (Q4) “The key pose was appropriately chosen”, and (Q5) “This figure needs more (manual) editing before I would share it with others.” The scale for Q1 was a 5-point Likert scale from “1: Strongly disagree” to “5: Strongly agree” and Q5 was from “1: Definitely needs edits” to “5: Very comfortable to share as is”. The answers for Q2 to Q4 were “Yes”, “No”, or “N/A”. Other comments and qualitative feedback were also collected.

### Study 2 Results

On average, participants completed task 1 in 5 mins with  $\mu = 2.3$  takes, task 2 in 10 mins ( $\mu = 2.8$  takes), task 3 in 3 mins ( $\mu = 2.3$  takes), and task 4 in 5 mins ( $\mu = 2$  takes). Figure A.2 and Figure A.3 provide examples of illustrations created by participants. Below we discuss participant feedback on generated illustrations and system design.

**Ratings and Accuracy.** Overall, participants thought the illustrations accurately described their motions (Q1 median ratings of 4.5 for task 1 and 4.88 for task 2). However, on average participants rated 4 of the 12 steps as requiring further editing to share with others (Q5 median ratings below 3 for both

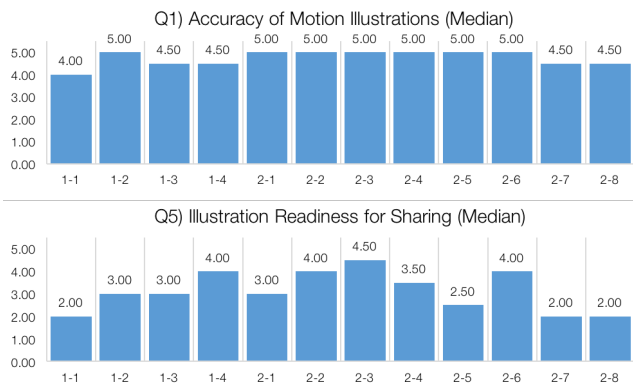


Figure 8. Study 2 median ratings for Q1 and Q5 by illustration step.

tasks). Figure 8 shows ratings for each step. Participants commented, “This figure represented the overall motion well (...) In particular, it captured all key poses, and the motion lines are easy to follow” (P8), but also “the system picked up really small movements in my other joints that were not relevant to the motion I was trying to depict (such as a small motion in my wrist or elbow)” (P1).

Across all 120 illustrations created all 10 participants, 99% showed all the important joints (Q2), and 80% precisely selected only the salient joints without extraneous movements (Q3). Participants commented: “the picture correctly represents my stance and body position. the arrows are easy to see and follow” (P1), “the lines were very accurate” (P2), and “the arcs are gorgeous and represent the intention of my motion really well” (P5).

Several participants appreciated how DemoDraw smoothed the motion arrows, especially when their demonstration was not perfect or there was capture noise. During the debrief session, participants were shown illustrations from different camera viewpoints and several noted the advantage: “I love the multiple camera angles for the wiggle arm motion I did in step 1” (P9).

**Key Pose Selection.** The answers to Q4 showed 94% of key poses were selected correctly. Example comments include: “The key poses are very descriptive of the motion” (P2), and “The key frames were just right” (P5).

**Authoring Workflow.** Participants found DemoDraw easy to learn (Median 5 out of 5) and easy to create illustrations with (Median 4.5). All participants were able to author and navigate using the speech interface. For example, “The voice command allows people be able to control the system remotely. Without the voice capability, the system can be impractical in the single-person use case” (P10).

Participants were especially impressed by how fast authoring could generate a step-by-step diagram: “Surprisingly fast to make some really cool full-body motion demonstrations. There is no way I could do this in higher-fidelity than a napkin sketch in the same time” (P5) and “the system saves significant amount of time creating illustrations” (P10). We also asked participants to estimate the time required if they were to gen-

erate a similar 8-step diagram without using DemoDraw, four participants answered that they would not be able to create them manually, while others responded that it would take 90 to 160 minutes based on each single figure taking 10-20 minutes.

**Improving a Demonstration.** The immediate visual feedback during the capturing phase was effective in helping authors review, adjust, and retake their performances. P4 explained, “I learned how to exaggerate the important aspects of motion without being explicitly told to.” In task 3, when we introduced the retaking capability, participants commented that this function would be especially helpful for a long motion sequence. All but one participant chose to retake step 2-7, which involved a holding position with one foot. Three participants later used the same technique for task 4. P8 also noted that it was helpful as “I could improve this by retaking that step and moving smoothly” when referring to a specific pose they thought needed additional work.

### Study 3: Refinement Interface Effectiveness

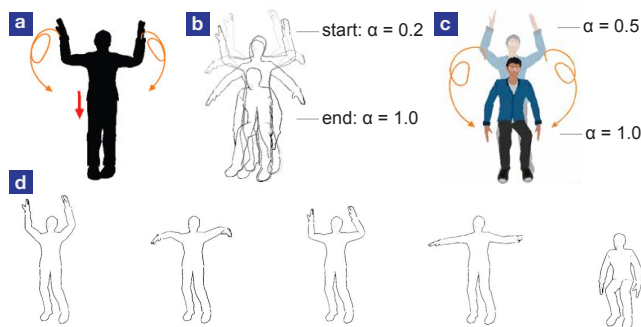
To understand how users refine automatically-generated results and generate different visualization styles in Refinement Interface, we conducted an informal study with 4 participants from an IT company (all males, aged 23 to 32 years, M=28). The same apparatus as Study 2 was used. First, experimenters guided participants during a 5-minute training phase to load one motion recording and create an illustration by manipulating a set of visual parameters. In the evaluation phase, participants were given two motion recordings, each with three illustrations created using our system. We presented the printed figures one by one and asked them to reproduce them. Participants then used DemoDraw to physically perform and record two specific motions given by the experimenter, and create illustrations to best convey each motion.

**Results.** Participants actively experimented with visual parameters and styles provided by the Refinement Interface, especially when creating stroboscopic effects. For example: number of intermediate frames and offset, dragging to reposition the arrows, and arrow color and width. In addition, participants formed strong preferences for styles once given these visualization options. For example, P4 said the stroboscopic effect in task 3 “is exactly what I looked for – It clearly conveys the start and end poses.” P2 preferred the cartoon renderer over the silhouette since “this character looks just like me!” All said they could not create similar illustrations without DemoDraw (Median 2 out of a 5-point Likert scale). This indicates detailed editing with the Refinement Interface was effective and expressive for various motion types.

### DISCUSSION

Participants were clearly excited about the overall experience using both the Demonstration Interface and the Refinement Interface. Some explicitly pointed out their enjoyment: “it accurately captures how much fun I had making it. :)” (Study 2-P9), and “For professional artists, the system not only increases their productivity, but also brings joy and fun to this kind of tasks” (Study 2-P10). Participant feedback suggests motion illustrations generated by DemoDraw are expressive





**Figure 9.** Different illustration effects conveying the same motion recording using DemoDraw’s Refinement Interface: a and c are created by the paper authors and a was used in Study 1; b by Study 3-P1 using 4 intermediate frames with zero offset; d by Study 3-P2 using 5 frames, positioned as a sequence.

enough to depict their demonstrations. Our multi-modal interface with motion analysis and rendering algorithms enabled users to quickly create step-by-step diagrams.

**Support of Various Styles.** In Study 1, motion arrows successfully conveyed the majority of movements. When arrows alone are not adequate, stroboscopic could be combined to clarify details in the start, intermediate, and end poses. For example, the simultaneous hand movements and squatting action of Step 3 in Study 1 are difficult to convey using only arrows (see Table 1), but Study 3 participants chose a stroboscopic effect to convey the same motion (see Figure 9). These findings align with existing instructional design principles [34], which suggest designers combine illustration styles based on the context. An authoring interface (like Refinement Interface in our system) must make it easy to adjust styles and parameters within a style (i.e., an outline figure with motion arrows).

**Iterative Creation Process.** Our studies verified our earlier findings that creating motion illustrations is an iterative procedure, where (re-)performing, reviewing, and refining are necessary components (see Figure 3). In Study 2, we observed how retaking a partial demonstration could be useful for long step-by-step motion sequences; Study 3 suggested that once moving into a refinement phase, authors focused on detailed adjustments of captured movements. Ensuring a high-level review of capturing results during demonstrations is therefore important.

**Animation vs. Illustration.** As DemoDraw captures the continuous motion sequence in 3D from a demonstrator, our system also generates animations showing the dynamic movements. In the warm-up task of Study 2 that captured the second motion set in Study 1, some participants explained that the playback animation of the recording clarified the motion where they incorrectly interpreted the start position. We propose that as motion arrows can efficiently and effectively express most of the motions, a mixed-media version can be created, where viewers can selectively review part of a static diagram with in-place animation playback. Such format has been shown to be useful for clarifying step-by-step instructions [14]. In addition, the 3D reconstruction also makes it possible to review motions from different viewing angles. All in all, our tech-

nology enables both instructors and viewers to interactively create and review motion illustrations in multiple ways.

**Design Implications.** In this work, we focus on body motion diagrams. More broadly, we believe our work also provides findings and techniques that apply to other demonstration-based systems. First, our multi-modal interface enables authors to perform critical tasks without leaving the demonstration context. Second, separating demonstration capture from detailed refinement allows users to focus solely on performance at demonstration-time while preserving the ability to fine-tune results later on. Third, clear, real-time illustration previews of performed motions enable fast iteration at demonstration.

## LIMITATIONS AND FUTURE WORK

Like any system, there are limitations imposed by architectural decisions and limits in available technology.

**Limited Interactions in Demonstration Mode.** Presently, authors can review and retake steps using voice commands, but many fine-grained operations are only available in the Refinement Interface, which requires users to leave the performance area. Future work should investigate if voice commands combined with gestures can expose more functionality like timeline scrubbing to the author, to tighten the feedback loop between performance, context setting, and depiction.

**Motion Capture and Segmentation.** First, the quality of DemoDraw illustrations is limited by the accuracy of motion capture data. Second, our segmentation algorithms currently assume that motions are separated by periods of inactivity, so we cannot yet capture and segment continuous motions that might be necessary, e.g., in different sports, where interruptions are not possible. Third, retargeting motion from a human performer to an avatar can introduce artifacts when skeletal geometry does not match. Future work could apply retargeting approaches from the computer graphics literature [22] or examine if it is feasible to automatically generate suitable avatars that match performers’ anatomy more closely.

**Movements Involving Objects and Multiple Users.** Many illustrations focus on motions while holding props (e.g., a tennis racket or baseball bat in sports) or the manipulation of objects (e.g., furniture assembly). We do not yet support such motions as the Kinect depth sensor we employed are limited to track skeletons. While the general case seems very hard, using techniques for recognizing objects in video based on a library of 3D models [30] appears promising. Furthermore, recent work has proposed fine-grained 3D tracking of humans and objects [21] and hands [39]. These techniques may reduce the numbers of retakes to obtain an artifact-free performance observed in our studies. Our current implementation is for single user, but we argue that it is possible to include multiple performers by loading and controlling additional avatar models, which would be especially useful in dancing.

**Interpretability of Motions.** DemoDraw can visualize the trajectories of multiple joints in a single image, but does not yet take the different timing of sub-motions into account. This can make illustrations of complex motions hard to interpret. Future work could provide per-joint timelines and automatically number sub-motions by their start times. In addition, the

dynamics of motion are not adequately represented in output images. To address this, we have begun to experiment with mixed-media output formats. Inspired by MixT [14], we can render static illustrations that can replay a motion segment as an animation when clicked.

## CONCLUSION

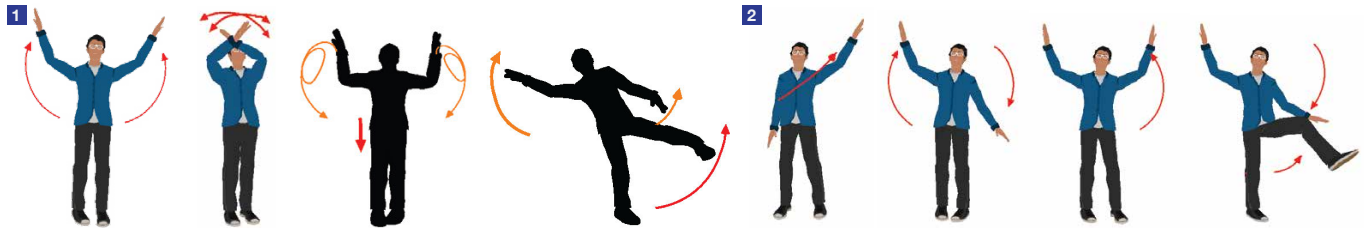
We introduced DemoDraw, a multi-modal system for generating human motion illustrations by physically demonstrating desired movements. It translates speech and 3D joint motion into a segmented sequence of key poses and salient joint movements, which are used to automatically generate a series of motion illustrations in effective and understood illustration styles. A multi-modal Demonstration Interface enables authors to record, review, and retake physical movements, and later refine and explore different motion visualizations with a Refinement Interface. We believe this “demonstrate-refine” pattern will generalize to other demonstration-based authoring systems. The primary motivation of this work is to provide users with domain-appropriate authoring tools that free them from tedious low-level tasks – allowing them to focus their effort on both communicative and aesthetic aspects. We look forward to applying the same approach to other instructional materials and illustration types in the future.

## REFERENCES

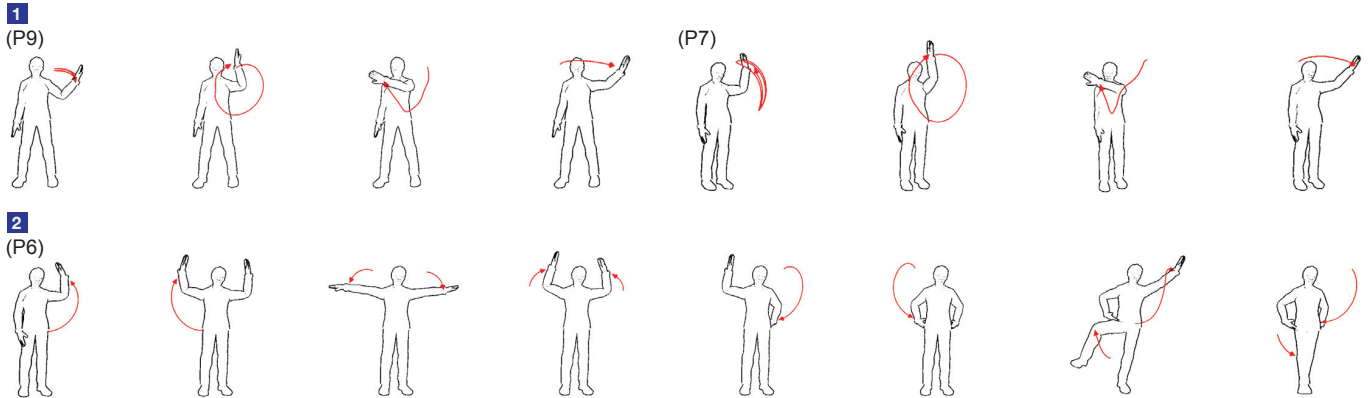
1. J.K. Aggarwal and M.S. Ryoo. 2011. Human Activity Analysis: A Review. *ACM Comput. Surv.* 43, 3, Article 16 (April 2011), 43 pages. DOI: <http://dx.doi.org/10.1145/1922649.1922653>
2. Maneesh Agrawala, Wilmot Li, and Floraine Berthouzou. 2011. Design Principles for Visual Communication. *Commun. ACM* 54, 4 (April 2011), 60–69. DOI: <http://dx.doi.org/10.1145/1924421.1924439>
3. Fraser Anderson, Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2013. YouMove: enhancing movement training with an augmented reality mirror. In *Proceedings of UIST '13*. ACM, 311–320.
4. Robert Anderson and Jean Anderson. 2010. Before and After Weight Training. In *Stretching*. Shelter Publications, Bolinas, California, 210–211.
5. Jackie Assa, Yaron Caspi, and Daniel Cohen-Or. 2005. Action synopsis: pose selection and illustration. *ACM Transactions on Graphics (TOG)* 24, 3 (2005), 667–676.
6. Jackie Assa, Daniel Cohen-Or, I-Cheng Yeh, Tong-Yee Lee, and others. 2008. Motion overview of human actions. In *ACM Transactions on Graphics (TOG)*, Vol. 27. ACM, 115.
7. Ronald M. Baecker. 1969. Picture-driven animation. In *Proceedings of the May 14-16, 1969, spring joint computer conference*. ACM, Boston, Massachusetts, 273–288. DOI: <http://dx.doi.org/10.1145/1476793.1476838>
8. Connelly Barnes, David E. Jacobs, Jason Sanders, Dan B Goldman, Szymon Rusinkiewicz, Adam Finkelstein, and Maneesh Agrawala. 2008. Video Puppetry: A Performative Interface for Cutout Animation. In *ACM SIGGRAPH Asia 2008*. ACM, New York, NY, USA, 124:1–124:9. DOI: <http://dx.doi.org/10.1145/1457515.1409077>
9. Lawrence Bergman, Vittorio Castelli, Tessa Lau, and Daniel Oblinger. 2005. DocWizards: A System for Authoring Follow-me Documentation Wizards. In *Proceedings of UIST '05*. ACM, New York, NY, USA, 191–200. DOI: <http://dx.doi.org/10.1145/1095034.1095067>
10. Richard A. Bolt. 1980. Put-that-there: Voice and Gesture at the Graphics Interface. In *Proceedings of SIGGRAPH '80*. ACM, New York, NY, USA, 262–270. DOI: <http://dx.doi.org/10.1145/800250.807503>
11. Simon Bouvier-Zappa, Victor Ostromoukhov, and Pierre Poulin. 2007. Motion cues for illustration of skeletal motion capture data. In *Proceedings of the 5th international symposium on Non-photorealistic animation and rendering*. ACM, 133–140.
12. Aaron Bryden, George Phillips Jr., and Michael Gleicher. 2012. Automated Illustration of Molecular Flexibility. *IEEE Transactions on Visualization and Computer Graphics* 18, 1 (Jan. 2012), 132–145. DOI: <http://dx.doi.org/10.1109/TVCG.2010.250>
13. Bill Buxton. 2007. *Sketching User Experiences: Getting the Design Right and the Right Design*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
14. Pei-Yu Chi, Sally Ahn, Amanda Ren, Mira Dontcheva, Wilmot Li, and Björn Hartmann. 2012. MixT: automatic generation of step-by-step mixed media tutorials. In *Proceedings of UIST '12*. ACM, 93–102.
15. Pei-Yu Chi, Joyce Liu, Jason Linder, Mira Dontcheva, Wilmot Li, and Bjoern Hartmann. 2013. Democut: generating concise instructional videos for physical demonstrations. In *Proceedings of UIST '13*. ACM, 141–150.
16. Myung Geol Choi, Kyungyong Yang, Takeo Igarashi, Jun Mitani, and Jehee Lee. 2012. Retrieval and visualization of human motion data via stick figures. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 2057–2065.
17. Gabe Cohn, Daniel Morris, Shwetak Patel, and Desney Tan. 2012. Humantenna: using the body as an antenna for real-time whole-body interaction. In *Proceedings of CHI '12*. ACM, 1901–1910.
18. Jonathan Corum. 2012. Drawing Science in Sign. (2012). <http://style.org/sign/>
19. James E. Cutting. 2002. Representing motion in a static image: constraints and parallels in art, science, and popular culture. *Perception* 31, 10 (2002), 1165–93. DOI: <http://dx.doi.org/10.1068/p3318>

20. Mira Dontcheva, Gary Yngve, and Zoran Popović. 2003. Layered Acting for Character Animation. In *ACM SIGGRAPH '03*. ACM, New York, NY, USA, 409–416. DOI: <http://dx.doi.org/10.1145/1201775.882285>
21. Mingsong Dou, Sameh Khamis, Yury Degtyarev, Philip Davidson, Sean Fanello, Adarsh Kowdle, Sergio Orts Escolano, Christoph Rhemann, David Kim, Jonathan Taylor, Pushmeet Kohli, Vladimir Tankovich, and Shahram Izadi. 2016. Fusion4D: Real-time Performance Capture of Challenging Scenes. In *ACM SIGGRAPH '16*.
22. Michael Gleicher. 1998. Retargetting motion to new characters. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. ACM, 33–42.
23. Dan B Goldman, Brian Curless, David Salesin, and Steven M Seitz. 2006. Schematic storyboarding for video visualization and editing. In *ACM Transactions on Graphics (TOG)*, Vol. 25. ACM, 862–871.
24. Amy Gooch, Bruce Gooch, Peter Shirley, and Elaine Cohen. 1998. A non-photorealistic lighting model for automatic technical illustration. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. ACM, 447–452.
25. Floraine Grabler, Maneesh Agrawala, Wilmot Li, Mira Dontcheva, and Takeo Igarashi. 2009. Generating photo manipulation tutorials by demonstration. *ACM Transactions on Graphics (TOG)* 28, 3 (2009), 66.
26. Saul Greenberg, Sheelagh Carpendale, Nicolai Marquardt, and Bill Buxton. 2012. *Sketching User Experiences: The Workbook*. Elsevier/Morgan Kaufmann. <https://books.google.com/books?id=c-RAUXk3gkC>
27. Ankit Gupta, Maneesh Agrawala, Brian Curless, and Michael Cohen. 2014. MotionMontage: A System to Annotate and Combine Motion Takes for 3D Animations. In *Proceedings of CHI '14 (CHI '14)*. ACM, New York, NY, USA, 2017–2026. DOI: <http://dx.doi.org/10.1145/2556288.2557218>
28. Ankit Gupta, Dieter Fox, Brian Curless, and Michael Cohen. 2012. DuploTrack: A Reatime System for Authoring and Guiding Duplo Model Assembly. In *Proceedings of UIST '12*. ACM, New York, NY, USA, 13.
29. Robert Held, Ankit Gupta, Brian Curless, and Maneesh Agrawala. 2012. 3D puppetry: a kinect-based interface for 3D animation.. In *Proceedings of UIST '12*. Citeseer, 423–434.
30. Natasha Kholgade, Tomas Simon, Alexei Efros, and Yaser Sheikh. 2014. 3D object manipulation in a single photograph using stock 3D models. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 127.
31. Wilmot Li, Maneesh Agrawala, Brian Curless, and David Salesin. 2008. Automated generation of interactive 3D exploded view diagrams. In *ACM Transactions on Graphics (TOG)*, Vol. 27. ACM, 101.
32. David Lumb. 2013. "Manga Generator" Uses The Kinect To Put Your Smooth Moves In A Custom Comic. (Sept. 2013). <http://www.fastcolabs.com/3016870>
33. Scott McCloud. 1994. *Understanding Comics* (reprint edition ed.). Avon, New York.
34. P. Mijksenaar and P. Westendorp. 1999. *Open here: the art of instructional design*. Joost Elffers Books. <https://books.google.com/books?id=fsJVAAAAMAAJ>
35. Niloy J Mitra, Yong-Liang Yang, Dong-Ming Yan, Wilmot Li, and Maneesh Agrawala. 2010. Illustrating how mechanical assemblies work. *ACM Transactions on Graphics-TOG* 29, 4 (2010), 58.
36. Peter Mohr, Bernhard Kerbl, Michael Donoser, Dieter Schmalstieg, and Denis Kalkofen. 2015. Retargeting Technical Documentation to Augmented Reality. In *Proceedings of CHI '15*. ACM, New York, NY, USA, 3337–3346. DOI: <http://dx.doi.org/10.1145/2702123.2702490>
37. Dorée Duncan Seligmann and Steven Feiner. 1991. Automated Generation of Intent-based 3D Illustrations. *SIGGRAPH Comput. Graph.* 25, 4 (July 1991), 123–132. DOI: <http://dx.doi.org/10.1145/127719.122732>
38. Richard Tang, Hesam Alizadeh, Anthony Tang, Scott Bateman, and Joaquim A.P. Jorge. 2014. Physio@Home: Design Explorations to Support Movement Guidance. In *CHI EA '14*. ACM, New York, NY, USA, 1651–1656. DOI: <http://dx.doi.org/10.1145/2559206.2581197>
39. Jonathan Taylor, Lucas Bordeaux, Thomas Cashman, Bob Corish, Cem Keskin, Eduardo Soto, David Sweeney, Julien Valentin, Benjamin Luff, Arran Topalian, Erroll Wood, Sameh Khamis, Pushmeet Kohli, Toby Sharp, Shahram Izadi, Richard Banks, Andrew Fitzgibbon, and Jamie Shotton. 2016. Efficient and Precise Interactive Hand Tracking through Joint, Continuous Optimization of Pose and Correspondences. In *ACM SIGGRAPH '16*.
40. Edward Tufte. 1990. *Layering and Separation*. Graphics Press, Cheshire, Connecticut.
41. Ross Yeager. 2013. *An Automated Physiotherapy Exercise Generator*. Master's thesis. EECS Department, University of California, Berkeley. <http://www.eecs.berkeley.edu/Pubs/TechRpts/2013/EECS-2013-91.html>
42. Yupeng Zhang, Teng Han, Zhimin Ren, Nobuyuki Umetani, Xin Tong, Yang Liu, Takaaki Shiratori, and Xiang Cao. 2013. BodyAvatar: Creating Freeform 3D Avatars Using First-person Body Gestures. In *Proceedings of UIST '13 (UIST '13)*. ACM, New York, NY, USA, 387–396. DOI: <http://dx.doi.org/10.1145/2501988.2502015>

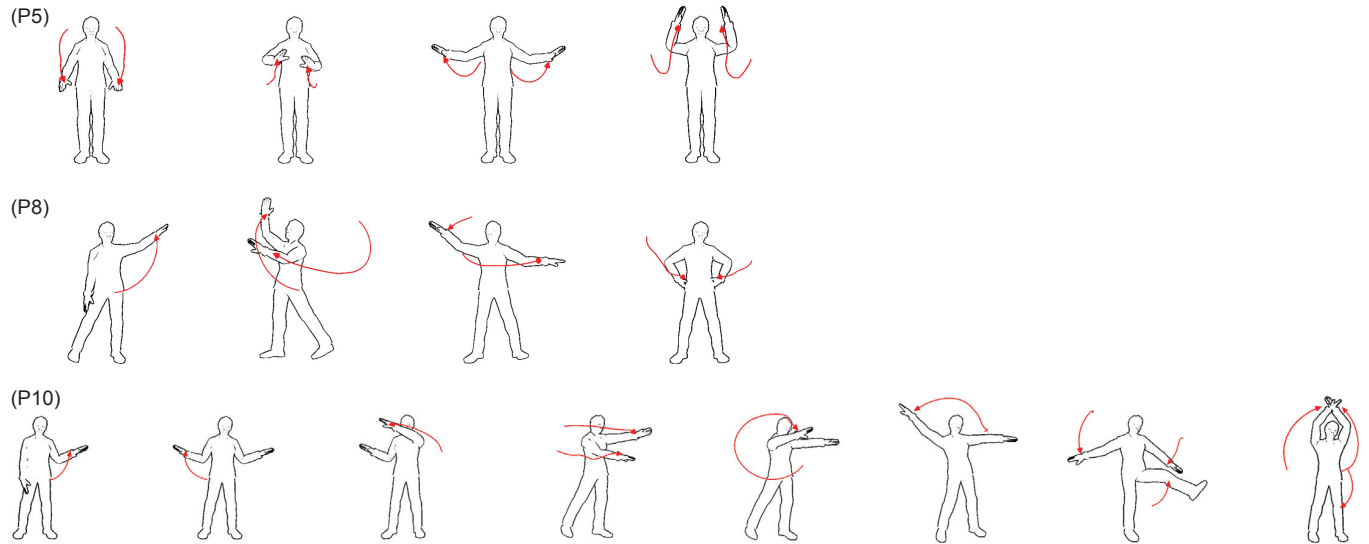
**APPENDIX**



**Figure A.1. Tasks provided in Study 1: We showed the printouts of these two sets of 4-step motions generated by DemoDraw using both the Demonstration Interface and the Refinement Interface. We asked participants to re-perform in front of a camera.**



**Figure A.2. Step-by-step illustrations generated by participants in Study 2 using the Demonstration Interface: 1) Results from P9 and P7 show the same four gestures of interface control in task 1, and 2) Results from P6 show 8-step moves in task 2.**



**Figure A.3. Selected illustrations from the open-ended task created by three different participants using the Demonstration Interface in Study 2: P5 performed to conduct a 4/4 beat pattern; P8 and P10 each performed four and eight free moves.**