

# Beat-based gesture recognition for non-secure, far-range, or obscured perception scenarios\*

**Graylin Trevor Jay**

Department of Computer Science  
Brown University

tjay@cs.brown.edu

**Patrick Beeson**

TRAC Labs Inc.  
Houston, TX

pbeeson@trac labs.com

**Odest Chadwicke Jenkins**

Department of Computer Science  
Brown University

cjenkins@cs.brown.edu

## Abstract

Gesture recognition is an important communication modality for a variety of human-robot applications, including mobile robotics and ambient intelligence domains. Most all gesture recognition systems focus on estimating the position of the arm with respect to the torso of a tracked human. As an alternative, we present a novel approach to gesture recognition that focuses on reliable detection of time-dependent, cyclic “beats” given by a human user. While the expressiveness of “beat-based” gestures is limited, beat-based gesture recognition has several benefits, including reliable 2D gesture detection at far ranges, gesture detection anywhere in the image frame, detection when the human is mostly hidden or obscured, and secure detection via randomly rotated beat patterns that are known only by the user and the perception system. In addition to discussing this complimentary approach to gesture recognition, we also overview a preliminary implementation of beat-based gestures, and demonstrate some initial successes.

## 1 Introduction

Gestures form the basis of most all non-verbal human communication. Thus, reliable gesture recognition is an important communication modality for a variety of human-robot applications [Kojo *et al.*, 2006; Jenkins *et al.*, 2007; Waldherr *et al.*, 2000], including mobile robotics and ambient intelligence domains. Gesture recognition can be used alone, or in conjunction with speech [Nicolescu and Mataric, 2003; Rybski *et al.*, 2007], to communicate spatial information, deliver commands, or update an intelligent observer on the status of the human. Pose-based gesture recognition techniques that estimate the position and orientation of the arms with respect to the torso have recently received a great deal of attention, especially depth-based efforts like Microsoft’s Kinect and other infrared systems [Knoop *et al.*, 2006].

The authors of this paper have successfully used depth-based pose estimation as the basis for gesture systems in the

past [Loper *et al.*, 2009]. These systems can perform well, as reinforced by our personal experience and by the recent popularity of the Kinect. However, there are practical and engineering disadvantages to such pose-based systems. There are many situations where gestures may be needed but where the human’s torso or arms may not be fully observable. Also, pose-estimation itself often requires the commitment of valuable resources. These resources may be computational, such as when the pose-estimation must be deduced from a monocular video stream, or may be physical, such as integrating the texture projection and dual infrared camera system of the Kinect or purchasing more expensive sensors for outdoor use.

In these situations, where practical or resource considerations make pose-estimation techniques less appropriate, it is ideal to have an alternative system that is less reliant on the need for accurate object recognition and modeling. We present just such an alternative approach that is based purely on motion, specifically on the detection of cyclic motions. Our “beat-based” system can reliably detect well-timed waving of the arm in a horizontal or vertical direction without the need for object recognition of any kind. While the expressiveness of “beat-based” gestures is limited compared to pose-estimation system, beat-based gesture recognition has several benefits. Beat-based gestures can be detected in a 2D image stream at near or far distances. Detection can occur anywhere in the image frame, which allows the human to be hidden, or allows attention to be focused on a particular location of the image. Also, by allowing the perception system to randomly change the required beat frequency, only a user with knowledge of the current beat will be able to signal the robot or sensor-equipped environment. These characteristics make beat-based recognition a natural complement to more traditional recognition systems that try to determine the exact posture and motion of a well-perceived human.

## 2 Motivation and Approach

While close-proximity, unobscured gesture recognition is possible in a large number of scenarios, many situations require a user to send (at least simple) signals to the robot in more obfuscated environments and at greater distances. For example, an operator may lead a robot to a particular location and ask it to begin patrolling. Later, the operator may wish to interrupt the patrol and ask the robot to “heel”. Ideally, the operator should be able to do this at the limits of the

\*This work was funded by U.S. Army Phase I STTR Contract #W56HZV-10-C-0443 (Topic: A10A-T030, “Leader-Following for Mobile Robots”).

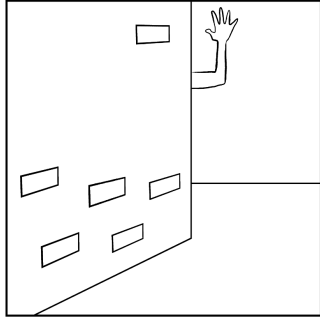


Figure 1: A motion-based gesture system allows for signals to be provided even if the user is hiding.

visual range of the robot. Another relevant example is that of a medic or bomb disposal soldier needing to signal an autonomous robot without exposing their body to an unsafe line of sight (see Figure 1). Similarly, an emergency worker may want to signal a robot that is partially occluded by debris, smoke, or fire.

Our attempts at beginning to address this problem have been guided by the observation that, in terms of difficulty for monocular perception: pose-estimation  $\gg$  recognition  $\gg$  feature detection. “Difficulty” here refers both to the computational and data/sensor resources required. For example, pose-estimation usually requires not only a fast system to perform real-time spatial modeling but also relatively high-resolution data. For a practical example, consider that even analog consumer cameras have long been built with the ability to be triggered by motion, as motion detection is simple. While most modern digital consumer cameras can perform rudimentary facial recognition to locate possible humans inside a photo, most consumer camera capabilities (as of this writing) do not yet have the computational ability to reliably track the orientation of bodies, arms, etc. As such, it would be ideal for camera systems to focus on regions and trigger upon detecting specific human-made motion signals. Such a system would always have greater range than pose detection (or even face detection) and would require fewer resources.

We were inspired by similar work commonly done with time-based signals [Carlotto, 2007]. It is very common to recognize artificial or human activity in time-based data by finding patterns of auto-correlation (i.e. cycles). These do not occur very often in nature. In adapting this approach to the 2D spatial domain we have come to focus on the detection of repetitive motion at specific frequencies. Thus, we call the resulting gestures “beat-based” gestures. By focusing on motion alone, there is no need for higher level recognition (e.g., of arms and hands), and motions involving relatively few pixels can be detected, which support ranges further than most 3D sensors can handle and at the effective range of most 2D sensors.

### 3 Implementation

Our beat-based gesture approach has been implemented in two distinct ways. In the first iteration, the software was completely correlation based, where changes in pixels were



Figure 2: The initial implementation, here used to launch and land a drone, was based on a correlation threshold between a supplied signal (a “beat”) and perceived motion.

correlated to know cycle times. In order to provide for more flexibility and to require less precise gestures, the more recent implementation is based on motion analysis similar to optical flow.

#### 3.1 Initial Implementation

The first iteration of our system was correlation based. A pre-defined oscillating signal, a beat, was given to a sensing program and to the operator (in the form of a flashing light). Motion, perceived as texture (i.e. pixel intensity) changes within the video stream was correlated with this signal. If the correlation reached a certain threshold, the pixels involved were considered to represent an executed gesture. To perform a gesture, the operator simply timed his or her movements to the flashing light.

In practice, this system was capable of a greater range of gestures than the simple binary set implied by the correlation threshold because depending on the type of movement, for example pendulation with the hand facing up versus down, different shaped patterns of pixels would reach the correlation threshold before others. These patterns could be distinguished from each other and this allowed for a simple two gesture take-off and land control of an AR Drone (see Figure 2).

In practical testing (e.g., in outdoor environments) the system required the operator to perform only five or six cycles of a gesture before detection. However, the high amount of user feedback needed was impractical and the need for strong synchronization in the system made implementation and parameter changes difficult. To address these problems the system was extended to allow for the detection of oscillations simply approaching (rather than being very accurately synced) with a target frequency at *any* phase. This required a more complex motion perception system.

### 3.2 Motion Perception

One of the traditional tools for motion analysis is optical flow [Lucas and Kanade, 1981]. Optical flow is often calculated by searching for a collection of per-pixel motion vectors that best explains the evolution of a series of images with respect to a number of constraints. Such constraints usually include: 1) that the changes in the pixel intensity values of an image sequence are due purely to motion, 2) that movements that change the relationship of a pixel to its neighbors (in terms of intensity values) are less likely than those that do not, 3) that a pixel is more likely to be involved in a motion similar to one its neighbors are involved in. In the nomenclature of the literature, these constraints are known as the optical flow constraint, the gradient constancy assumption, and the smoothness constraint. Typically, assumptions and constraints of optical flow are treated as costs and the set of best motion vectors is discovered by solving a minimization/optimization problem. Solving such an optimization problem can be quite computationally intense and is often not possible in real-time without specialized hardware or programming techniques.

Our motivation for having a motion-focused system is that motion is a salient and local feature. Many of the constraints of optical flow, when taken together, are equivalent to a non-local analysis of image structure. For example, many fast approximations of “true” optical flow are implemented in terms of tracking higher-level features such as corners. To make our motion analysis as local as possible and to avoid the computational complexity of optical flow, we implemented our own simpler alternative based on a single assumption/constraint: All detected texture changes are due to the motion of persistent objects. In the implementation, this assumption is applied by keeping a record of the time since each pixel experienced a texture (i.e. pixel intensity) change greater than a set threshold. When the assumption above holds, gradients within the resulting value field capture the direction of the motion of objects (see Figure 3).

On top of this motion perception facility, we implemented a visual oscillation detector that could be tuned to a particular frequency. We adapted a state machine approach similar to the one often used in visual synchronization. Initial results have been encouraging, with low overhead for accurate detection. In this state machine approach every pixel is assigned a counter and a timer. For horizontal detection, when a leftward motion is detected the timer is reset. When a rightward motion is detected the current value of the timer is examined. If the timer’s value is close (by a pre-determined tolerance) to the amount that would be expected given the oscillation frequency being sought out, the counter is incremented. Pixels whose counter exceeds a threshold are considered to be involved in oscillations at the target frequency. The system detects a gesture when a target number of pixels are actively involved in such oscillations.<sup>1</sup> Because we classify motion as

<sup>1</sup>One advantage to this entire approach is that all operations are performed on a per-pixel basis except for the final counting of oscillating pixels and the calculation of the local motion gradients; however, even in these cases a very limited number of neighboring pixels are involved.

horizontal or vertical based on the motion gradients, oscillations based on horizontal motion can be detected as distinct from oscillations made by vertical motion.

It is important to note that the beat-based gesture detection is based the frequency of *alternating* left-right or up-down motions and *not* simply on the frequency of motion. The system is thus only sensitive to properly oscillating motions and not other forms of cyclic motion. The system would not, for example, respond to a blinking light, even if it was blinking at the target frequency, nor would it respond to a rotating object whose period of rotation was the target frequency.

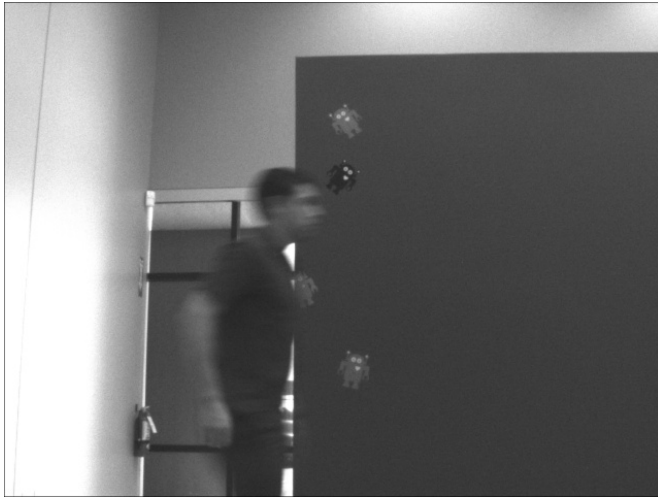
To test this algorithm’s reliability in far-range, outdoor situations, we tuned the system to trigger a detection after seeing three consecutive back-and-forth oscillations at 1 Hz. The oscillations could occur at any location in the 640×480 images of the motionless camera. Even with this low-resolution camera, we were able to achieve a working distance of ~25.6 meters (~84 ft) with highly reliable detection of 1 Hz arm gestures and with no false positives (even with palm trees blowing and vehicles driving in the distance). We did notice a reduction in quality whenever the background has a similar intensity to the human’s arm. Wearing dark sleeves against a light background (or light sleeves against a dark background) can overcome this issue in the initial implementation. Figure 4 shows images from this testing, where the camera used for beat-based gestures is mounted on a mobile robot base.

### 3.3 Increasing Expressiveness

Although the system could demonstrate highly reliable oscillation detection at far ranges using a still camera, our beat-based recognition suffered from occasional false positives when mounted on mobile platforms operating in highly dynamic environments. These false positives were overcome in two ways. First, whenever the camera itself is moving (known by monitoring any pan-tilt devices or the robot base), the gesture software ignores the incoming camera frames. This reduces false positives caused by ~1 Hz oscillations due to panning or tilting of the camera back and forth. This reduces many false positives, but also means that the robot must be perfectly still when beat-based gestures are given. This is accomplished by having an adequate “dead-zone” for any actuation (pan-tilt, zoom, or mobile base control) during tracking behaviors.

To further reduce any false positives and to extend the expressiveness of the system, we decided to augment the system with human tracking information. Given the rough location of the tracked human, the 2D camera image is divided into regions of interest that correspond to areas to the left and right of the operator’s torso (see Figure 5(a) and <http://www.youtube.com/watch?v=83S6x7g8lMM>). Regions of interest reduce false positive detections caused by a number of factors—most often the human’s own body rocking back and forth during conversation or other natural activities. By estimating the distance/scale of the person, the regions of interest grow and shrink as the leader moves closer and nearer to the camera. This has the added benefit of allowing the beat-based software to be more sensitive when the leader is further away, resulting in improved performance at far distances.

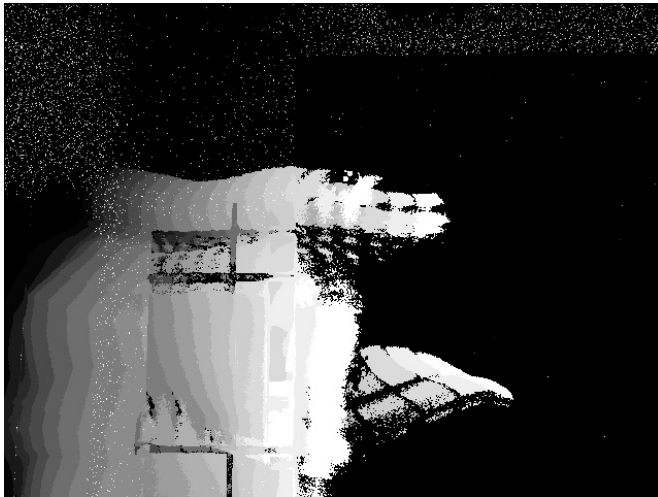
In addition to reducing false positives and increasing op-



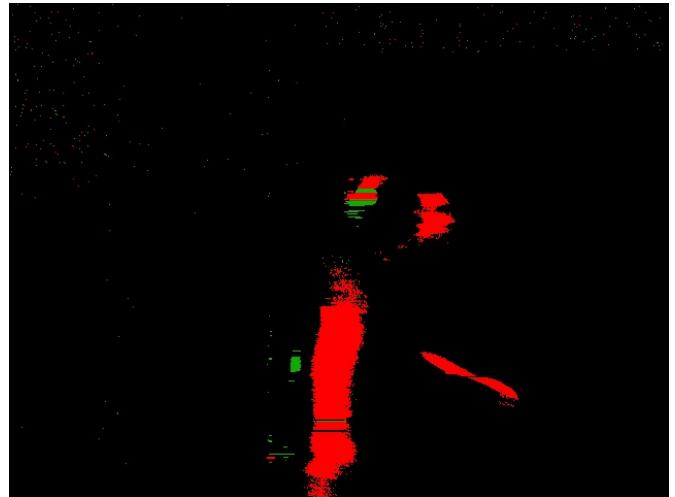
(a)



(b)



(c)



(d)

Figure 3: (a,b) Motion causes temporal changes in pixel values. (c) A “trail” image is created recording which pixels were active changing over time (brighter indicates more recently changed pixels). (d) Determining the gradient of a pixel in the trail image can be used to determine its direction of motion. Here, the high degree of red corresponds to a rightward gradient and indicates the person must be moving rightward.



(a)



(b)



(c)

Figure 4: (a) Even with 640x480 resolution images, the beat-based gestures work well at far distances. (b) An example image from the camera at the maximum distance where gestures work reliably. Notice the small number of pixels that fall onto the user's right arm. (c) When the background intensity (grayscale) is similar to the user's arm intensity (left arm in image), the arm motion cannot be separated from the background, resulting in a failure. Future efforts will focus on overcoming these threshold problems.

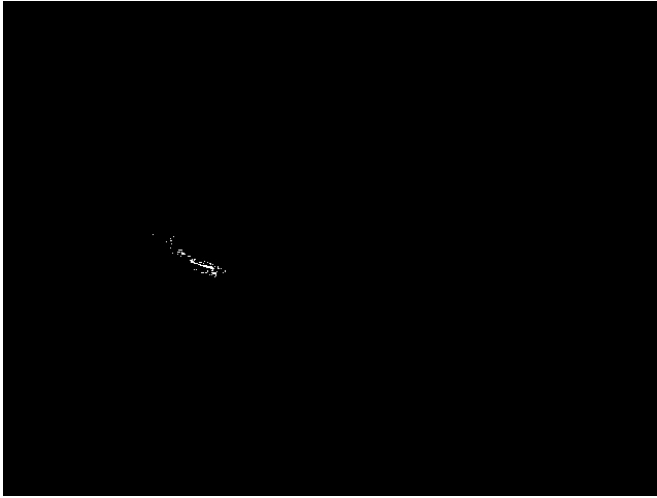




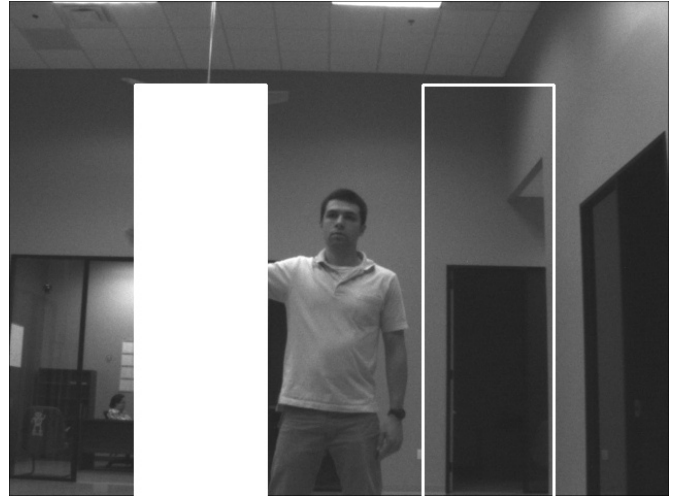
(a)



(b)



(c)



(d)

Figure 5: Utilizing human tracking information, right and left arm oscillations can be distinguished. (a-b) An oscillation of the right arm is performed. (c) The final frame of the trace is used to classify the location of the oscillation with respect to the tracked individual. (d) This yields left/right classification of 1 Hz arm oscillations, with no observed incorrect classifications to date.

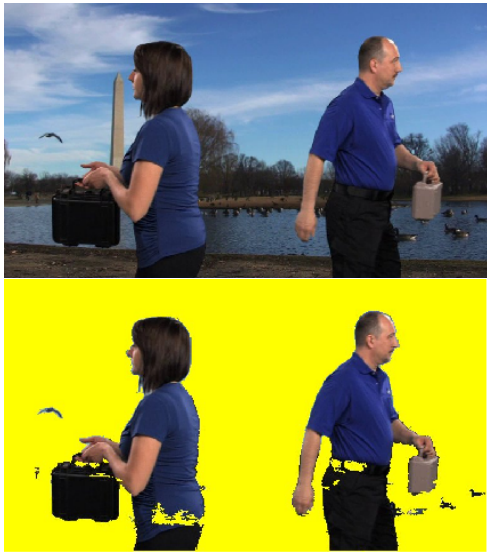


Figure 6: Using the method proposed by KadewTraKuPong and Bowden [2001], small repetitive motions (like leaves or water) are ignored by constantly adapting a mixture of Gaussians to recent frames. Large motions across many pixels are quickly determined via adaptive background subtraction.

erating range, knowing the leader location also allows us to distinguish beat gestures made with the left or right arm (Figure 5), giving us 4 signals (left/vertical, left/horizontal, right/vertical, right/horizontal). Videos showing left/right arm beat-based gesture recognition working both indoors and outdoors and turning on and off different tracking and following behaviors can be seen at <http://www.youtube.com/watch?v=55F928QVXOI>.

As it was available on the mobile platform, the human tracking used for augmentation of the system was depth-based. Specifically, a custom 3D template matching approach. As can be seen in the accompanying videos, none of the resulting 3D human modeling data was used for recognition. As such, the low level of tracking accuracy required to distinguish left from right arm beat gestures makes it very likely that an alternative, less resource-intensive, approach such as facial recognition, traditional people-following [Schulz, 2006; Gockley *et al.*, 2007], texture-based tracking, or even sound or radio-based localization could have been used to achieve similar results.

## 4 Future Work

As mentioned, we have noticed that the intensity thresholds used for real-time motion detection might not always distinguish the human’s arm from backgrounds of similar intensity. In the future, will focus on refining the algorithms to increase the reliability of far-range gesture recognition and to recognize more complex motions. First, we will use more principled approaches to segmenting the moving portions of the image stream from the static background. Rather than using static thresholds, we can use thresholds that quickly adapt to particular environment and lighting conditions. KaewTraKuPong and Bowden [2001] present a method that we

have used in the past to quickly and reliably pick out motions from image sequences while ignoring sensor noise (see Figure 6). Similarly, we plan to create alternative methods for motion perception, such as adding an additional smoothness constraint, which should eliminate sensitivity to spurious or unintended movement. This would allow beat-based gesture detection to compensate for camera movement, from pan/tilt actuation or from the base rotation/translation.

Next, we will investigate a larger variety of beat-based motions. Our initial implementation cannot distinguish between oscillations at the target frequency (typically 1 Hz in testing) and higher frequency harmonics. This means that providing a beat gesture at a higher multiple of the desired frequency (e.g., at 2 or 3 Hz) is equivalent to providing the gesture at the desired frequency; however, it may be beneficial to allow beat gestures at 2 Hz or 3 Hz to mean different things. Categorizing the exact frequency of beat-based gestures would allow for the same basic motions performed at different speeds to be assigned different semantics.

Additionally, we would like to continue to utilize the simplifying assumption that all detected texture changes are due to the motion of persistent objects. In the past this was only true when the camera was motionless. In the future, we should be able to leverage our tight perception/control loop in order to remove motion between frames caused by pan-tilt or mobile base actuation. This way, beat-based gestures can be detected even on continuously moving platforms, like unmanned aerial vehicles.

Finally, we would like to further explore some of the implications of our initial correlation-based approach. Potentially, when based on a shared signal, beat-based gestures could provide highly secure visual communication between a leader and the robot. Suppose instead of always looking for a 1 Hz cyclic gesture, the robot only acknowledges motions that synchronize with a randomly determined beat, and suppose this beat changes periodically, much like the login information of a secure key-fob. If the operator has access to this information (e.g., via a headset tuned to an encrypted signal or simply through a synchronized clock), then only the operator can command the robot, as only the leader will know the current beat required to control the robot. Such a scenario could be used to acquire a robot’s attention, to change between operators, and to keep a threat from taking control of the robot. It would also allow for multiple operator/robot pairs to operate at once in a shared space.

In the context of our larger work, which focuses on gesture-controlled human-following systems, we hope to utilize beat-based gestures to develop a hybrid-approach to far-range 2D visual tracking. The approach will utilize well accepted approaches to motion-based tracking [Dang *et al.*, 2002] and cutting-edge work in attention-based recognition [Mishra *et al.*, 2009]. Attention-based techniques attempt to address the “chicken and egg” problem of background separation—that it is easy to separate a recognized object from a background or to recognize an object properly separated from the background, but that either is difficult to perform alone with a natural image. Attention-based techniques solve this problem by performing segmentation based on an assumed object center. In previous work, these object centers were supplied by

an operator or some other system (such as stereo disparity); however, due to its ability to perceive motions without object recognition, the current motion perception system presents a unique opportunity to obtain these centers directly.

## 5 Conclusion

We have presented our experiences around the implementation and use of a purely motion-sensitive, beat-based gesture system. We believe that these experiences indicate great promise for the use of such approaches in hybrid systems integrating a large number of interaction modalities [Stiefelhagen *et al.*, 2004; Rogalla *et al.*, 2002; Haasch *et al.*, 2004; Kennedy *et al.*, 2007] or in applications where pose-based systems are either infeasible, impractical, or overly expensive. We hope to further extend our experimental systems in the future, and we look forward to seeing others with more theoretical experience in motion modeling studying this particular approach and its areas of application.

## Acknowledgments

The authors would like to thank Ben Conrad for his help in creating the images and videos that demonstrate this work.

## References

- [Carlotto, 2007] M. J. Carlotto. Detecting patterns of a technological intelligence in remotely sensed imagery. *Journal of the British Interplanetary Society*, 60:28–30, 2007.
- [Dang *et al.*, 2002] T. Dang, C. Hoffmann, and C. Stiller. Fusing optical flow and stereo disparity for object tracking. In *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, pages 112–117, September 2002.
- [Gockley *et al.*, 2007] R. Gockley, J. Forlizzi, and R. G. Simmons. Natural person-following behavior for social robots. In *Proceedings of the ACM/IEEE International Conference of Human-Robot Interaction*, pages 17–24, March 2007.
- [Haasch *et al.*, 2004] A. Haasch, S. Hohenner, S. Huwel, M. Kleinhagenbrock, S. Lang, I. Tóptsis, G. Fink, J. Fritsch, B. Wrede, and G. Sagerer. BIRON – the Bielefeld robot companion. In *Proceedings of the International Workshop on Advances in Service Robotics*, pages 27–32, June 2004.
- [Jenkins *et al.*, 2007] O.C. Jenkins, G. Gonzalez, and M.M. Loper. Interactive human pose and action recognition using dynamical motion primitives. *International Journal of Humanoid Robotics*, 4(2):365–385, June 2007.
- [KaewTraKulPong and Bowden, 2001] P. KaewTraKulPong and R. Bowden. An improved adaptive background mixture model for real-time tracking with shadow detection. In *Proceedings of the European Workshop on Advanced Video-Based Surveillance Systems*, September 2001.
- [Kennedy *et al.*, 2007] W. G. Kennedy, M. Bugajska, M. Marge, W. Adams, B. R. Fransen, D. Perzanowski, A. C. Schultz, and J. G. Trafton. Spatial representation and reasoning for human-robot collaboration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1554–1559, July 2007.
- [Knoop *et al.*, 2006] S. Knoop, S. Vacek, and R. Dillmann. Sensor fusion for 3D human body tracking with an articulated 3D body model. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1686–1691, May 2006.
- [Kojo *et al.*, 2006] N. Kojo, T. Inamura, K. Okada, and M. Inaba. Gesture recognition for humanoids using proto-symbol space. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robots*, pages 76–81, December 2006.
- [Loper *et al.*, 2009] M. Loper, N. Koenig, S. Chernova, O. Jenkins, and C. Jones. Mobile human-robot teaming with environmental tolerance. In *Proceedings of the ACM/IEEE International Conference of Human-Robot Interaction*, pages 157–164, March 2009.
- [Lucas and Kanade, 1981] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of Imaging Understanding Workshop*, pages 121–130, 1981.
- [Mishra *et al.*, 2009] A. Mishra, Y. Aloimonos, and C. Fermüller. Active segmentation for robotics. In *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, 2009.
- [Nicolescu and Mataric, 2003] M. N. Nicolescu and M. J. Mataric. Natural methods for robot task learning: Instructive demonstrations, generalization and practice. In *Proceedings of the International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 241–248, July 2003.
- [Rogalla *et al.*, 2002] O. Rogalla, M. Ehrenmann, R. Zollner, R. Becher, and R. Dillmann. Using gesture and speech control for commanding a robot assistant. In *Proceedings of the IEEE International Workshop on Robot and Human Interactive Communication*, pages 454–459, September 2002.
- [Rybski *et al.*, 2007] P. E. Rybski, K. Yoon, J. Stolarz, and M. Veloso. Interactive robot task training through dialog and demonstration. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction*, pages 49–56, March 2007.
- [Schulz, 2006] D. Schulz. A probabilistic exemplar approach to combine laser and vision for person tracking. In *Proceedings of the Robotics: Science and Systems Conference*, August 2006.
- [Stiefelhagen *et al.*, 2004] R. Stiefelhagen, C. Fugen, R. Giesemann, H. Holzapfel, K. Nickel, and A. Waibel. Natural human-robot interaction using speech, head pose and gestures. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 2422–2427, October 2004.
- [Waldherr *et al.*, 2000] S. Waldherr, S. Thrun, and R. Romero. A gesture-based interface for human-robot interaction. *Autonomous Robots*, 9(2):151–173, 2000.