Soli: Ubiquitous Gesture Sensing with Millimeter Wave Radar

Jaime Lien,* Nicholas Gillian, M. Emre Karagozler, Patrick Amihood, Carsten Schwesig, Erik Olson, Hakim Raja, Ivan Poupyrev

Google ATAP

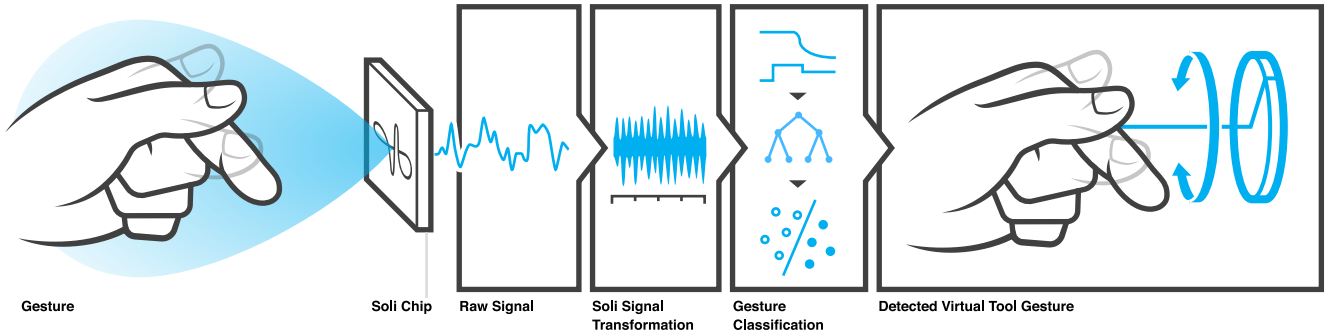


Figure 1: *Soli is the first millimeter-wave radar system designed end-to-end for ubiquitous and intuitive fine gesture interaction.*

Abstract

This paper presents *Soli*, a new, robust, high-resolution, low-power, miniature gesture sensing technology for human-computer interaction based on millimeter-wave radar. We describe a new approach to developing a radar-based sensor optimized for human-computer interaction, building the sensor architecture from the ground up with the inclusion of radar design principles, high temporal resolution gesture tracking, a hardware abstraction layer (HAL), a solid-state radar chip and system architecture, interaction models and gesture vocabularies, and gesture recognition. We demonstrate that Soli can be used for robust gesture recognition and can track gestures with sub-millimeter accuracy, running at over 10,000 frames per second on embedded hardware.

Keywords: sensors, interaction, gestures, RF, radar

Concepts: •Human-centered computing → Interaction devices; Gestural input; •Hardware → Sensor devices and platforms;

1 Introduction

This paper presents *Soli*, a new, robust, high-resolution, low-power, miniature gesture sensing technology for interactive computer graphics based on millimeter-wave radar. Radar operates on the principle of reflection and detection of radio frequency (RF) electromagnetic waves [Skolnik 1962]. The RF spectrum has several highly attractive properties as a sensing modality for interactive systems and applications: the sensors do not depend on lighting, noise or atmospheric conditions; are extremely fast and highly precise; and can work through materials, which allows them to be easily embedded into devices and environments. When implemented at millimeter-wave RF frequencies, the entire sensor can be designed as a compact solid-state semiconductor device: a *radar chip* that is a miniature, low-power device having no moving parts and can be manufactured inexpensively at scale. The resulting Soli sensor delivers the promise of truly ubiquitous gesture interaction across a very broad range of applications, including but not limited to virtual reality (VR), wearables and smart garments, Internet of Things (IoT) and game controllers, as well as more traditional devices such as mobile phones, tablets and laptops.

It is important to point out that the first radar systems were developed as early as the 1930s [Watson-Watt 1945], and that RF sensing has since become a well established and mature field of engineering and applied science. The current radar hardware and computational methods, however, were primarily developed for mainstream radar applications, which usually involve detection and tracking of large moving objects at large distances, such as air and terrestrial traffic control, marine radar, aircraft anti-collision systems and outer space surveillance, and geophysical monitoring, among many others. The engineering requirements for such applications are not compatible with modern consumer applications in which sensors must fit into tiny mobile and wearable devices, run on limited computational resources, work at ultra-short distances (i.e. less than 5 mm), consume little power, and track the dynamic configuration of complex, highly deformable elastic objects, such as a human hand as opposed to a rigid airplane, at sub-millimeter accuracy. We are not aware of any existing radar system that could satisfy the above requirements. Our investigation suggests that developing a radar-based sensor optimized for human-computer interaction (HCI) requires re-thinking and re-building the entire sensor architecture from the ground up, starting with basic principles.

In this work, we present the first end-to-end radar sensing system specifically designed for tracking and recognizing fine hand gestures. Our work builds upon a large existing body of knowledge in the radar domain and, for the first time, explores the comprehensive design principles, implementation, and optimization of these tools for scalable gesture sensing within the constraints and requirements of modern HCI. We show that ubiquitous and intuitive gesture in-

*This research was supported by a Multi-University Research Agreement with Google while the author was a student at Stanford University.

SIGGRAPH 2016, July 24-28, 2016, Anaheim, CA
ISBN: 978-1-4503-ABCD-E/16/07
DOI: http://doi.acm.org/10.1145/9999997.9999999

ACM Reference Format
Lien, J., Gillian, N., Karagozler, M., Amihood, P., Schwesig, C., Olson, E., Raja, H., Poupyrev, I. 2016. Soli: Ubiquitous Gesture Sensing with Millimeter Wave Radar. ACM Trans. Graph. 35, 4, Article 142 (July 2016), 19 pages. DOI = 10.1145/2897824.2925953 http://doi.acm.org/10.1145/2897824.2925953.

teraction is made possible through tailored, interdependent design of the entire sensor architecture, from the radar sensing paradigm (Sections 3 and 4) to hardware and solid-state radar chips (Section 5), interaction models (Section 6), gesture tracking techniques and vocabularies (Figure 1 and Section 7), and software pipelines (Section 8). The complete end-to-end design, development, and evaluation of this new gesture sensing modality is the major achievement and contribution of this work, opening new research frontiers in non-imaging sensors for interaction.

2 Background and Related Work

Input is an essential, necessary and critical component of interactive computer graphic systems. Exploration of novel input devices and modes of interaction to allow an operator to control computer generated objects stretches back decades to the first pen interfaces by Ivan Southerland, the computer mouse by Engelbart, the "Put it there" gesture interface by Bolt, PHIGS and GKS, Nintendo Power Glove, GOMS and early multitouch capacitive screens by Lee and Buxton, etc. [Sutherland et al. 1963; Bolt 1980; Shuey et al. 1986; Barnes 1997; Lee and Buxton 1985; Card et al. 1983]. Despite significant progress, the importance of creating new input devices has not diminished; it remains a highly relevant, active and growing area of research both in academia and industry.

The development of new input technologies is closely connected to the changing context of the use of computing. One of the most important developments of the last decade is the explosive growth of mobile computing, where mobile phones, tablets, hand-held game devices, wearables and re-emerging VR have grown into a dominant platform to access and interact with information. Indeed, as of 2014, there are more people accessing content using mobile devices than with desktop computers [Comscore Inc. 2014]. As mobile computing grows, new modes of interaction are emerging and becoming feasible, including touch and touchless gestures using camera-based tracking or capacitive field sensors, voice and gaze input, and a multitude of sensors embedded in various objects, the human body, clothes and environments or distributed as 3D interfaces (see for example [Kim et al. 2012; Harrison et al. 2010; Rekimoto 2001; Saponas et al. 2009; Smith et al. 1998; Strickon and Paradiso 1998; Dietz and Leigh 2001; Russell et al. 2005; Gustafson et al. 2011; Holz and Wilson 2011; Yatani and Truong 2012; Cooperstock et al. 1997; Holleis et al. 2008; Chan et al. 2015].) The choice of particular input technologies is always application driven and involves difficult tradeoffs between size, power consumption, ease of integration into the devices and environment, sensitivity to light and environmental noise, cost, update rate and precision of tracking, and many other considerations.

With Soli, we were motivated by the emergence of applications in wearable, mobile and ubiquitous computing where the traditional touch interaction is difficult if not impossible, and where free air gestures emerge as a promising and attractive form of human-computer interaction [Song et al. 2014; Kim et al. 2012]. In particular we are interested in gestures that involve highly precise and controlled motions performed by small muscle groups in the wrist and fingers. It has been well established over decades of research that small muscle groups in the hands allow for fluid, effective and rapid manipulation, resulting in precise and intuitive interaction [Zhai et al. 1996]. The superior capabilities of fingers and small muscle groups are particularly evident when we observe extremely accurate and fluid expert interaction with hand tools, such as those used by a watch maker. Creating input technology that can capture similar precision of finger motion with free touchless gestures was one of the important motivations of our work on the Soli sensor.

Designing interfaces that would allow the capture of precise and fast hand motion in free space has proven to be challenging. Computer vision techniques have achieved great progress over the last decade [Kim et al. 2012] but still suffer from latency and difficulty resolving overlapped fingers and objects, requiring instrumentation of human hands [Wang and Popović 2009; Dorfmuller-Ulhaas and Schmalstieg 2001]. Using depth information by employing either multiple cameras or a single depth camera can significantly improve precision [Sharp et al. 2015], but does not scale well to the small form factor required by wearable and mobile devices. Capacitive and field sensing scales well across applications, but does not allow for high definition interaction in free space [Smith et al. 1998].

In this paper we explore the use of a new physical medium for fine gesture interaction: millimeter-wave radar. Although radar has been an active area of research and development since the 1940s, its use in interactive systems has been episodic and rare. This is probably due to the large size, high power consumption and significant computational demand required to generate and process radar signals until recently. As the first miniature impulse radars were developed in the 1990s [Azevedo and McEwan 1996] and radar technology has become more accessible, driven primarily by the proliferation of radar in automotive applications, there have been a few explorations of radar for presence detection, recognizing walking patterns, and detecting breathing and sleep patterns [Rahman et al. 2015; Zhuang et al. 2015; Otero 2005; Wang and Fathy 2011].

Existing work in RF gesture sensing is limited to tracking rigid single targets such as single limb or body motion [Paradiso 1999; Paradiso et al. 1997; Pu et al. 2013; Kellogg et al. 2014], large scale hand motion [Wan et al. 2014; Molchanov et al. 2015c], and a pen [Wei and Zhang 2015]. We are not aware of any previous work that attempted to use RF sensors to precisely track fine motions of multiple fingers and recognize dynamic gestures expressed by complex deforming hand configurations. As noted in [Wei and Zhang 2015], this problem is fundamentally different from tracking the whole hand and presents unique challenges when the radar sensor resolution is limited by application constraints such as size, power, and computational load. In particular, we explore for the first time *scalable and ubiquitous* gesture interaction systems made possible by modern millimeter-wave radars, which allow for compact design, precise tracking at close range, and can be manufactured inexpensively and at large volume.

The Soli radar sensor that we report in this paper has many applications that extend far beyond hand gestures, and opens many avenues for designing exciting and effective user interfaces in various fields of computing. In the current work however, we focus on gesture interaction with applications in wearable, mobile computing as well as in the emerging fields of IoT and ubiquitous computing.

3 Radar Fundamentals

The fundamental principles of radar sensing are straightforward (see Figure 2). A modulated electromagnetic wave is emitted toward a moving or static target that scatters the transmitted radiation, with some portion of energy redirected back toward the radar where it is intercepted by the receiving antenna. The time delay, phase or frequency shift, and amplitude attenuation capture rich information about the target's properties, such as *distance*, *velocity*, *size*, *shape*, *surface smoothness*, *material*, and *orientation*, among others. Thus these properties may be extracted and estimated by appropriately processing the received signal.

The *goal of radar system design* is to optimize radar functional performance for the specified application, such as gesture tracking in the case of Soli, within the application's constraints. The design of any radar system includes a) *hardware*, such as antennas and internal circuitry components, b) *signal processing* techniques to modu-

late the transmitted waveform and extract information from the received waveform, and c) radar *control software* that executes radar operation and algorithms [Richards et al. 2010]. The design of all these elements is strongly interconnected and cannot be specified independently from each other or the specifics of the application.

Historically, radar system design was driven by applications such as detecting, locating, and identifying aircraft, ships, and other rigid targets [Skolnik 1962; Brown 1999]. These applications are significantly different from tracking and recognizing complex, dynamic, deforming hand shapes and fine motions at very close range. In the rest of this section, we briefly review existing radar techniques and discuss challenges for applying them to the new realm of mobile gesture sensing. We follow with a discussion of Soli radar design principles in Section 4. For a comprehensive review of radar, we refer interested readers to [Skolnik 1962; Richards et al. 2010].

3.1 Modeling radar targets

In traditional tracking radar systems, a target is modeled and abstracted by a single parameter called *radar cross section* (RCS), which represents its ability to reflect electromagnetic energy back toward the radar [Knott 2012]. Note that RCS is a property of the target, not of the radar system, and does not directly correspond to the target's actual physical cross section.

In gesture tracking applications where the hand is very close to the sensor, abstracting the hand as a single radar cross section yields extremely coarse discriminatory information, determined primarily by the physical cross-section of its pose. For fine gesture recognition, where overlapped motions of the individual figures must be captured, RCS is insufficient. Thus, instead of using RCS, we model the hand as a collection of *dynamic scattering centers*, as described in the next section.

3.2 Range resolution of the radar

The classic approach for target identification with radar is to map its measured reflectivity and dynamics to spatial coordinates [Smith and Goggans 1993]. This technique relies on spatially resolving multiple parts of the target. The *range resolution* of a radar system refers to the minimum physical separation in the radar's line-of-sight between two *distinguishable* points. We note this metric is different from *accuracy* or *precision*, which refer to the radar's ability to measure the distance to a single point. The classic equation that defines the radar's spatial resolution in the range dimension is as follows:

$$res_r = \frac{c}{2BW}, \tag{1}$$

where c is the speed of light and BW is the total bandwidth of the transmitted waveform. Given that the largest bandwidth currently allowed by FCC is 7 GHz in the 60 GHz band [FCC 2016], the finest possible radar range resolution is approximately 2 cm. Clearly this resolution is inferior to other gesture tracking techniques, for example, see Microsoft Kinect sensor resolution measurements [Khoshelham and Elberink 2012].

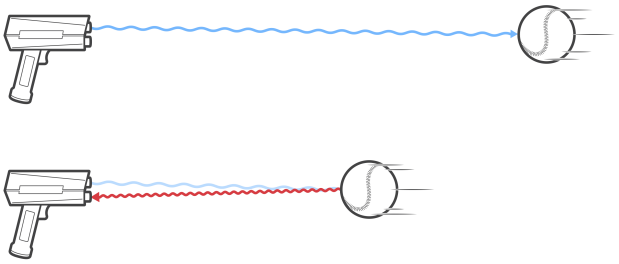


Figure 2: *The fundamental principles of radar sensing are based on transmission and reflection of RF waves.*

In order to overcome this limitation, we develop a novel paradigm for radar sensing that eschews high spatial resolution for extremely fine temporal resolution. This paradigm, which applies to any form of signal modulation, is described in the next section.

3.3 Beam steering and beam width

By steering the radar beam either mechanically or digitally using electronically steered antenna arrays, the radar can track not only distance but also the target's location in horizontal and vertical dimensions, e.g. in range and azimuth [Brookner 1985]. Beam steering has been used to recover target shape [Ralston et al. 2010]; thus the brute force approach to gesture tracking would suggest designing a narrow pencil beam and *scanning it across the hand* to spatially resolve its skeletal structure and individual finger motion.

The challenge of using this classic technique for gesture sensing is that it would require designing a very large radar antenna to achieve the necessary angular resolution. Indeed, the angular resolution at range r can be computed as [Skolnik 1962]:

$$res_a = rb = \frac{r\lambda}{l}, \tag{2}$$

where b is antenna beam width, λ is the wavelength and l is the aperture size. Assuming that 1 cm angular resolution is needed to discriminate hand poses, a 60 GHz radar at a 20 cm distance would require an antenna aperture of 10x10 cm. This size is obviously prohibitive for small interaction sensors meant to be used in wearable and mobile applications. Moreover, the heavy computational burden required to infer the hand's skeletal structure would not be possible on power-constrained devices such as a smartwatch.

In Soli, we propose a novel approach where a broad antenna beam illuminates the entire hand, capturing a complex reflected signal that describes specific hand configurations. By using signal processing and machine learning techniques we can identify the corresponding hand gestures without spatial hand reconstruction.

3.4 Radar modulation and hardware agnostic design

The radar center frequency f_c has critical implications for its system design. Because antenna size is roughly proportional to wavelength, lower center frequencies require much larger antenna apertures. The choice of wavelength also defines RF propagation and scattering characteristics, as well as FCC regulatory requirements [FCC 2016]. Radar waveforms must then be *modulated* onto the center frequency to enable ranging functionality and resolution. Indeed, an unmodulated single tone radar such as a speed gun cannot measure distance because there is no reference for determining the time delay between transmitted and received waveforms. A wide variety of modulation techniques may be used, including *pulse modulation* [Hussain 1998], *frequency modulation* [Stove 1992], *phase modulation* [Levanon 2000], and *pseudo-noise modulation* [Axelsson 2004].

The choice and design of modulation schemes for specific applications are driven by multiple factors. These include the hardware and algorithmic complexity required to modulate the transmitted waveform and process the received signal, transmission duration, peak transmitted power, power efficiency, minimum and maximum detectable range, and susceptibility to various types of noise and

interference. Therefore, there is no modulation technique that is universally optimal for every application and environment. For this reason, a radar system has traditionally been designed as *a single purpose sensor*, highly optimized for its intended application both in hardware and software.

This high degree of specialization and integration between radar software and hardware makes radar less scalable to broad consumer market applications. To make radar a viable technology for ubiquitous use, the *sensing algorithms and software must be abstracted* from the low-level hardware details as well as specific signal modulation schemes. This abstraction allows running the *same application software* on hardware from various manufacturers, which can then be optimized for specific fields of use. In other words, as with all software stacks, we would like the application-level software to be independent from the hardware architecture and low-level signal processing specific to the hardware architecture.

The concept of a *hardware abstraction layer* (HAL) is well established in the sensor domain (e.g. cameras), but thus far has not been implemented for radar. In this paper we propose a set of abstractions that are universal across all radar architectures. We demonstrate this new radar HAL by implementing and running a single piece of gesture sensing software on three different radar architectures: FMCW, phase modulated spread spectrum, and impulse radar all operating at different center frequencies.

3.5 Solid-state radar devices

Electronic hardware design for high frequency radar can be challenging, requiring highly specialized equipment and skill. In particular, antenna and waveguide design for wideband, super-GHz frequencies can present a significant barrier to cost-efficient, ubiquitous sensing. We overcome this challenge by designing an all-in-one radar IC that integrates all radar functionality onto a single chip, including antennas and preprocessing that interface directly to a standard microprocessor that can be found in a normal mobile phone or a smart watch.

4 Soli Radar Design Principles

In this section, we present the fundamental design principles of the Soli radar. These principles circumvent the limitations of traditional techniques by employing new paradigms of radar sensing, specifically designed to track fine hand gestures in consumer applications.

The *overall approach* behind Soli radar design can be understood on an intuitive level as follows (see Figure 3). We illuminate the hand with a broad 150 degree radar beam with pulses repeated at very high frequency (1-10 kHz). The reflected signal is a superposition of reflections from multiple dynamic scattering centers that represent dynamic hand configurations. We then process the received signal into multiple abstract representations (that we call *transformations*), which allow us to extract various *instantaneous* and *dynamic* characteristics of the moving hand and its parts, which we call *features*. These features are insufficient to reconstruct the skeletal structure of the hand; however, their combinations can uniquely identify various hand configurations and motions that we can identify by comparing them to a priori captured sets of training data using machine learning techniques. In combination, these steps form the *Soli Processing Pipeline* (Soli PP). In the rest of this section we dive into specific details of various parts of the Soli PP.

4.1 Scattering center model of human hand

We model the RF response of the hand as a superposition of responses from discrete, dynamic scattering centers (see Figure 3 above). Scattering center models are consistent with the geometrical theory of diffraction when the wavelength is small in comparison to the target's spatial extent [Keller 1962], an assumption that holds for millimeter-wave sensing of the hand. In Equation 3 below, we propose a generalized time-varying scattering center model that accounts for non-rigid hand dynamics.

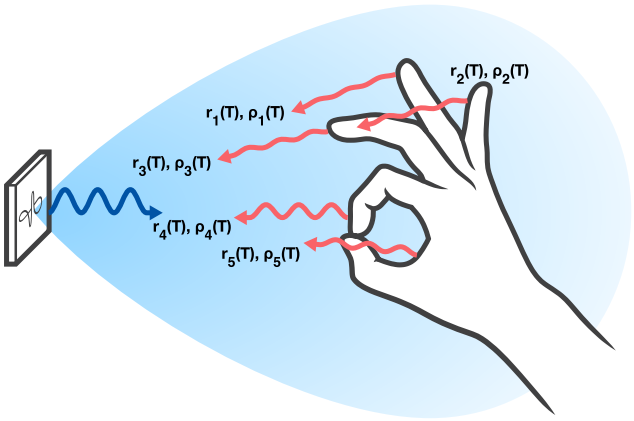


Figure 3: *Soli illuminates the entire hand and measures a superposition of reflections from multiple dynamic scattering centers.*

Each scattering center is parameterized by *complex reflectivity parameter* $\rho_i(T)$ and *radial distance* $r_i(T)$ from the sensor, which vary as a function of time T:

$$y(r,T) = \sum_{i=1}^{N_{SC}} \rho_i(T)\delta(r - r_i(T)), \tag{3}$$

where N_{SC} is the number of scattering centers and $\delta(.)$ is the Dirac delta function. [1] The complex reflectivity parameter ρ is frequency-dependent and varies with the local hand geometry, orientation with respect to the radar, surface texture, and material composition. This parametric description of the millimeter-wave scattering response for a dynamically reconfiguring hand presents a tractable model for gesture parameter estimation and tracking.

4.2 High temporal resolution

Unlike classic radar techniques that rely on high spatial resolution for target discrimination, Soli proposes a sensing paradigm based on *high temporal resolution*. In this paradigm we detect subtle and complex hand motions and gestures by measuring the hand's response to radar at extremely high frame rates, then extracting *fine temporal signal variations* corresponding to those hand motions and gestures. We implement this concept by transmitting a periodic modulated waveform,

$$s_{tr}(t,T) = u(t-T)\exp(j2\pi f_c t), T = 0, RRI, 2RRI, ..., \tag{4}$$

where f_c is the center frequency, $u(t)$ is the complex envelope defining one period of the modulation scheme, and RRI is the *radar repetition interval* indicating the time from the start of one modulation period to the start of the next. For each transmission period, a corresponding received waveform $s_{rec}(t,T)$ is measured. This transmission scheme defines *two distinct time scales* with which to analyze the reflected hand signals.

[1] We note that the linear relationship between time delay and range allows Equation 3 to be expressed in terms of either variable; with appropriate scaling, they are interchangeable for practical purposes.

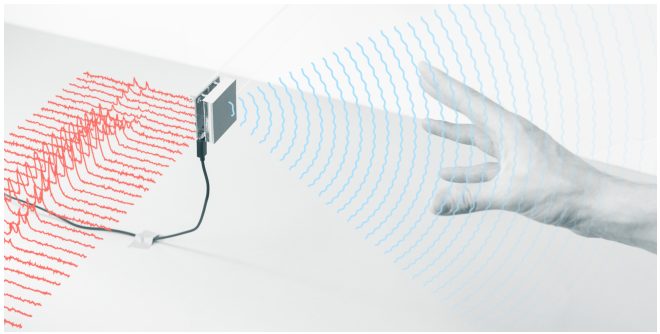


Figure 4: *The Soli received signal is measured as a function of slow time and fast time.*

In *slow time*, denoted by T, received signals are collected at the *radar repetition frequency* of $RRF = \frac{1}{RRI}$. For Soli radar, RRF varies between 1 kHz and 10 kHz. Within a single modulation period RRI, the transmitted waveform varies as a function of *fast time*, denoted by t, and the corresponding received signal is sampled at the analog-to-digital sampling rate (Figure 4).

High radar repetition frequency is the *fundamental design paradigm* linking the scattering center hand model to the Soli signal processing approach. For sufficiently high radar repetition frequency, we can assume that the scattering center properties are approximately constant over a single radar repetition interval due to the relatively slow motion of the human hand. Consequently, the scattering center range and reflectivity vary only as a function of slow time T, as indicated above in Equation 3.

The received radar signal within one repetition interval provides information about *instantaneous* scattering center range and reflectivity, while variations in the received radar signal over multiple repetition intervals result from scattering center *dynamics*; for example, velocity and change in geometry. We therefore extract characteristics describing the hand's instantaneous pose and orientation by processing $s_{rec}(t,T)$ as a function of fast time t, while tracking gesture dynamics by processing as a function of slow time T as shown in Figure 5. The hand motions and characteristics that we extract in fast and slow time are then used to estimate hand gestures. We discuss some of these characteristics below.

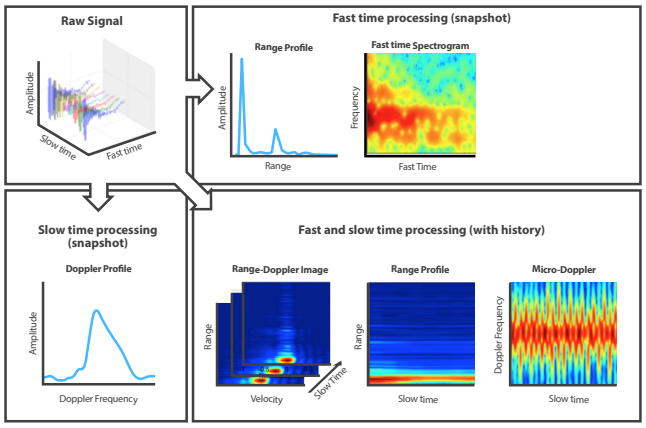


Figure 5: *The combination of slow time and fast time processing produces several signal transformations, including the range-Doppler, range profile, Doppler profile, and spectrogram.*

4.3 Wide antenna beam, fast time processing

Rather than designing a narrow antenna beam to spatially resolve the hand's scattering centers, the Soli radar *illuminates the entire hand in a single wide beam* during each transmission period. Because all scattering centers on the hand are simultaneously illuminated, the measured waveform $s_{raw}(t,T)$ consists of reflections from each scattering center, superimposed in fast time:

$$s_{raw}(t,T) = s_{tr}(t,T) * y(\frac{ct}{2},T) = \sum_{i=1}^{N_{SC}} s_{i,raw}(t,T). \quad (5)$$

Each individual reflected waveform $s_{i,raw}(t,T)$ is modulated by the associated scattering center's instantaneous reflectivity $\rho_i(T)$ and range $r_i(T)$ (see Figure 3 above) [2]:

$$s_{i,raw}(t,T) = \frac{\rho_i(T)}{r_i^4(T)} u(t - \frac{2r_i(T)}{c}) \exp(j2\pi f_c(t - \frac{2r_i(T)}{c})). \quad (6)$$

We note that the $\frac{1}{r^4}$ path loss model is idealized, but because we do not rely on received signal strength for distance measurement, our pipeline is not sensitive to inaccuracies in this model.

After RF demodulation and modulation-specific filtering (for example, matched filtering or pulse compression), the preprocessed received signal represents a superposition of responses $s_{i,rec}(t,T)$ from each of the hand's scattering centers:

$$s_{rec}(t,T) = \sum_i s_{i,rec}(t,T). \quad (7)$$

The response from the i-th scattering center is given by

$$s_{i,rec}(t,T) = \frac{\rho_i(T)}{r_i^4(T)} \exp(j\frac{4\pi r_i(T)}{\lambda}) h(t - \frac{2r_i(T)}{c}), \quad (8)$$

where λ is the wavelength within the propagation medium, and $h(t)$ is the radar system point target response. The shape of the point target response function determines the range resolution and depends on the particular modulation scheme, transmission parameters, and preprocessing steps.

Equations 7-8 show that the instantaneous properties $r_i(T)$ and $\rho_i(T)$ of all scattering centers $i = 1,\ldots,N_{SC}$ are *contained in the received signal* $s_{rec}(t,T)$. Each scattering center produces a point target response $h(t)$ that is delayed in fast time by the round-trip propagation time, scaled in amplitude by the scattering center's reflectivity $\rho_i(T)$, and modulated in phase by the scattering center's range $r_i(T)$. If two scattering centers i and j are separated in range by a distance greater than the range resolution, i.e.

$$|r_i(T) - r_j(T)| > \frac{c}{2BW}, \quad (9)$$

then their responses $s_{i,rec}(t,T)$ and $s_{j,rec}(t,T)$ are resolvable in the fast time dimension.

A variety of characteristics for the hand as a whole can also be extracted in fast time. For example, RF scattering properties and electromagnetic characteristics of the human hand can provide rich information about the hand shape, size, pose, texture, and any covering material [Baum et al. 1991]. In order to analyze the RF spectral

[2] For brevity, we ignore amplitude scaling factors that do not depend explicitly on scattering center parameters, e.g. antenna gain.

response of hand, we compute a fast time-frequency spectrogram decomposition of the raw received signal as follows (see Figure 5):

$$SP(t, f, T) = \int_{t}^{t+t_{win}} s_{raw}(t', T) \exp(-j2\pi f t') dt'. \quad (10)$$

For Soli's relaxed range resolution constraints, the properties and dynamics of each *individual scattering center* on the hand, e.g. speed or displacement, are usually not immediately resolvable in fast time. Instead, Soli relies on fine motion resolution to resolve the individual scattering center responses in slow time.

4.4 Fine motion resolution, slow time processing

High radar repetition frequency enables Soli to capture fine phase changes in the received signal $s_{rec}(t, T)$ corresponding to scattering center dynamics over slow time. These motion-induced temporal signal changes are used to *resolve the individual scattering center responses* from their complex superposition.

Soli's exceptional motion sensitivity results from the phase of each scattering center response $s_{i,rec}(t, T)$. As scattering center i moves from $r_i(T_1)$ to $r_i(T_2)$, the relative displacement produces a corresponding phase shift $\Delta\phi_i(T_1, T_2)$ proportional to λ:

$$\Delta\phi_i(T_1, T_2) = \frac{4\pi}{\lambda}(r_i(T_2) - r_i(T_1)) \mod 2\pi. \quad (11)$$

At 60 GHz, this relationship dictates that a range displacement of 1 mm produces 0.8π radians of phase change. With an observable phase jitter of about 0.25π radians, the Soli radar can theoretically measure displacement of an ideal point target with up to *0.3 mm accuracy*, which is very high when compared to other sensors.

The phase change dependence on displacement enables Soli to resolve scattering centers in slow time based on their phase histories. We assume that the velocity $v_i(T)$ of each scattering center i is approximately constant over some coherent processing time T_{cpi} greater than the radar repetition interval, i.e.

$$\frac{dr_i(T)}{dT} = v_i(T) \approx v_i \text{ for } T_{cpi} > RRI. \quad (12)$$

The phase history over the coherent processing time then produces a Doppler frequency

$$f_{D,i}(T) = \frac{1}{2\pi}\frac{d\phi_i(T)}{dT} = \frac{2v_i(T)}{\lambda}. \quad (13)$$

The Doppler frequencies of multiple scattering centers moving at different velocities can thus be resolved by computing the spectrum of $s_{rec}(t, T)$ in each fast time bin over the coherent processing slow time window T_{cpi}:

$$S(t, f, T) = \int_{T}^{T+T_{cpi}} s_{rec}(t, T') \exp(-j2\pi f T') dT'. \quad (14)$$

Functionally, this is achieved by buffering $N_{cpi} = T_{cpi} RRF$ consecutive preprocessed radar returns $s_{rec}(t, T)$ in a fast time versus slow time array, and then applying an FFT to each fast time row across the slow time columns. The resulting fast time-frequency mapping $S(t, f, T)$ is easily converted to range and velocity using the transformation

$$RD(r, v, T) = S(\frac{2r}{c}, \frac{2v}{\lambda}, T). \quad (15)$$

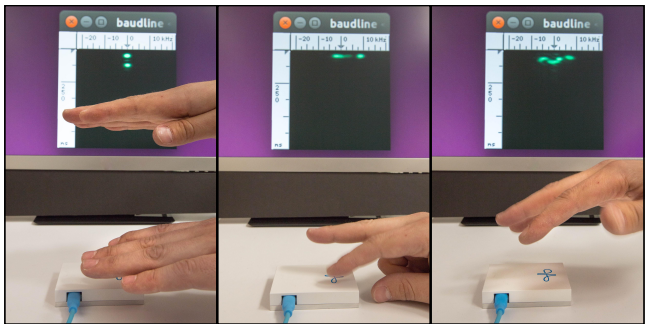


Figure 6: *Scattering center positions and dynamics can be resolved and differentiated in the range-Doppler signal transformation. (Left) Two stationary hands resolvable in range. (Middle) Two fingers moving at different velocities within the range resolution limit but resolvable in velocity. (Right) Multiple fingers moving at different resolvable velocities.*

The signal processing parameters T_{cpi} and RRF can be finely tuned for the expected hand dynamics and desired sensing performance in terms of SNR, velocity resolution, and Doppler aliasing.

The resulting three-dimensional range-Doppler array $RD(r, v, T)$ maps the reflected energy from each scattering center to its range $r_i(T)$ and velocity $v_i(T)$ at time T. The energy return from distinct scattering centers are thus resolvable if *any* of the following criteria are met (see Figure 6 above for examples):

1. their *separation in range* is greater than the range resolution, determined by $\frac{c}{2BW}$,
2. their *difference in velocity* is greater than the Doppler velocity resolution, determined by $\frac{\lambda}{2T_{cpi}}$, or
3. they are detectable only in *disjoint coherent processing time windows*.

It is important to emphasize that the Soli sensing paradigm does not require large bandwidth or high spatial resolution for gesture tracking. In fact, the achievable spatial resolution for a wearable radar form factor is coarser than the scale of most fine finger gestures, hence Criterion 1 above rarely applies. Instead, our fundamental sensing principles rely mostly on motion-based resolution, analyzing scattering center dynamics by processing temporal changes in the raw signal over slow time.

It is worth noting that range-Doppler processing is an initial step for many synthetic aperture radar (SAR) and inverse SAR rigid target imaging and discrimination techniques [Park and Kim 2010]. While an SAR imaging pipeline can be built on top of the Soli range-Doppler signal processing, we believe this approach is currently inappropriate for computationally constrained, real time gesture sensing applications. Not only is the rigid target assumption invalid for hand gestures with fine finger motions, but the algorithms required to infer spatial structure from Doppler measurements add significant computational overhead. We show that this additional processing is unnecessary for fine gesture recognition by *extracting and tracking gesture signatures directly from the motion space*.

4.5 Gesture motion profiles

Unlike image-based gesture sensors, Soli directly captures gesture motion profiles and temporal variations in scattering center properties as motion signatures for gesture recognition. Rather than deducing the hand's skeletal structure or spatial orientation of fingers, these motion patterns themselves can be input to machine learn-

ing techniques, along with more traditional features encoding the instantaneous hand properties. This approach reduces the computational load, making real time radar-based gesture recognition feasible even on low power devices such as smart watches and IoT.

The scattering center dynamics are captured by the *location and trajectory* of scattering center responses in the range-Doppler mapping, as well as its lower dimensional projections: the range profile,

$$RP(r,T) = \sum_v RD(r,v,T) \tag{16}$$

and Doppler profile, also known as micro-Doppler,

$$DP(v,T) = \sum_r RD(r,v,T). \tag{17}$$

The range profile and Doppler profile show the distribution of reflected energy over distance and velocity, respectively, as a function of slow time. These signals thus provide rich visualizations of hand dynamics over the temporal course of a full gesture.

4.6 Feature extraction

From the signal transformations described above, we extract a variety of low-dimensional features loosely classified as follows:

- explicit scattering center tracking features,
- low level descriptors of the physical RF measurement, and
- data-centric machine learning features.

The combination of these techniques allows Soli to capture gesture dynamics as well as instantaneous properties of the hand in low-dimensional features. We describe some of these features below.

4.6.1 Explicit tracking of scattering centers

Soli achieves robust hand and finger detection and tracking when the relevant scattering centers are resolvable according to the criteria in Section 4.4. We utilize constant false alarm rate (CFAR) detection to isolate individual scattering center responses in the range-Doppler image and track the trajectories of these responses over time. For each resolvable scattering center i we can continuously estimate its *range* $r_i(T)$, i.e. the radial distance from the sensor, *velocity* $v_i(T)$ and *acceleration* $\frac{dv_i(T)}{dT}$ in the radial direction, as well as *reflectivity* $\rho_i(T)$, i.e. a measure of energy reflected from the scattering center and intercepted by the radar (see Equations 3 and 12). Finally, a *number of resolvable scattering centers* $\tilde{N}_{SC}(T)$ can also be extracted.

4.6.2 Low level descriptors of RF measurement

Fundamental radar sensing and resolution limits often prevent the multiple fingers' scattering centers from being fully distinguishable in the range-Doppler image. For fine, fast, and fluid gestures, individual scattering centers are frequently unresolvable in velocity because the fingers are moving at similar speeds or their response is blurred over multiple Doppler bins due to acceleration during the coherent processing interval. We found that in such cases, low level abstract features characterizing the energy distribution across the signal transformation space can sufficiently describe relative finger dynamics.

Velocity profile centroid: The centroid of the range-gated Doppler profile correlates well with the relative motion of fingers:

$$v_{centroid}(T) = \sum_{r=r_0}^{r_1} \sum_v vRD(r,v,T) \tag{18}$$

where the boundaries r_0 and r_1 are determined by simple range estimation or hand tracking techniques described above.

Relative displacement: Relative finger displacement from time T_1 to T_2 can be estimated by integrating the range-gated velocity centroid:

$$disp(T_1,T_2) = \sum_{T=T_1}^{T_2} v_{centroid}(T). \tag{19}$$

Velocity profile dispersion: The dispersion of energy over the Doppler space describes the distribution of velocities during the coherent processing interval:

$$v_{dispersion}(T) = \sqrt{\frac{\sum_{r=r_0}^{r_1} \sum_v RD(r,v,T)(v - v_{centroid}(T))^2}{\sum_{r=r_0}^{r_1} \sum_v RD(r,v,T)}}. \tag{20}$$

Range profile dispersion: The energy spread over the range space describes the spatial extent of the targets in the radar's field of view.

$$r_{dispersion}(T) = \sqrt{\frac{\sum_r RP(r,T)(r - r_{i^*}(T))^2}{\sum_r RP(r,T)}}, \tag{21}$$

where i^* is the index of the resolvable scattering center with maximum reflectivity:

$$i^* = \arg\max_{i=1}^{\tilde{N}_{SC}} \rho_i(T). \tag{22}$$

Total instantaneous energy: The total energy of the received signal encodes various properties of the targets:

$$E_{instantaneous}(T) = \sum_r RP(r,T). \tag{23}$$

Total time-varying energy: The energy in the coherent difference between the received signal from frame to frame provides a measure of target fluctuation and movement.

$$E_{time-varying}(T) = \sum_r |\tilde{RP}(r,T) - \tilde{RP}(r,T-1)|. \tag{24}$$

where $\tilde{RP}(r,T)$ is the complex-valued range profile.

4.6.3 Data-centric machine learning features

A number of features are extracted directly from the Soli transformation data specifically for the purpose of input to the Soli machine learning algorithms (see Section 7). Prior to computing data-centric features, the transformations data is normalized to the unit scale and a region of interest (ROI) is defined in Soli transformations. For example, that could mean tracking the main energy centroid in the normalized range-Doppler transformation and extracting a rectangular ROI that is centered on the range bin of the energy centroid. Using ROI significantly reduces the dimensionality of the search over using the entire transformation data. Some of the following data-centric features are then extracted within this ROI.

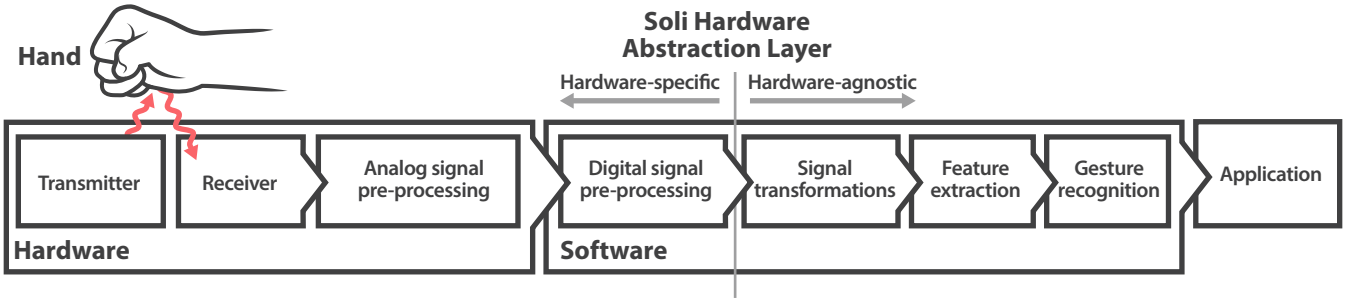


Figure 7: *The Soli Processing Pipeline implements algorithmic stages of increasing data abstraction from raw radar signal to application-specific gesture labels.*

Range Doppler Multichannel Integration (RDMI): The RDMI matrix combines data from the ROIs of K virtual channels. To further reduce the dimensionality, the RDMI data can be downsampled, resulting in an $\frac{1}{M_{ds}} ROI_{\text{rows}}, \frac{1}{N_{ds}} ROI_{\text{cols}}$ matrix. Each element in this matrix at time T is given by:

$$\mathbf{RDMI}_{ij}^{(T)} = \frac{1}{M_{ds} N_{ds} K} \sum_{m=1}^{M_{ds}} \sum_{n=1}^{N_{ds}} \sum_{k=1}^{K} RD_{rc}^{(T,k)}, \tag{25}$$

where $\text{RDMI}_{ij}^{(T)}$ is the i'th row and j'th column of the RDMI matrix at time T, and $RD_{rc}^{(T,k)}$ represents the r'th row and c'th column from the k'th channel of range Doppler from the Soli transformations at time T. Here r and c are given by: $r = x + i * M_{ds} + m$, and $c = j * N_{ds} + n$, where x is an offset to ensure the range-Doppler centroid is in the center of the ROI. M_{ds} and N_{ds} control the downsample factor for the rows and columns respectively.

Range Doppler Multichannel Derivative (RDMD): The RDMD matrix extracts differences between two channels of range Doppler:

$$\mathbf{RDMD}_{ij}^{(T)} = \frac{1}{M_{ds} N_{ds}} \sum_{m=1}^{M_{ds}} \sum_{n=1}^{N_{ds}} \left[RD_{rc}^{(T,0)} - RD_{rc}^{(T,1)} \right] \frac{1}{\delta T}, \tag{26}$$

where δT represents the sample rate of the system to normalize the derivative across varying sample rates.

Range Doppler Temporal Derivative (RDTD): The RDTD matrix extracts changes in the RDMI matrices over two consecutive time steps, which describes how the signal is changing over time:

$$\mathbf{RDTD}_{ij}^{(T)} = \left[\text{RDMI}_{ij}^{(T)} - \text{RDMI}_{ij}^{(T-1)} \right] \frac{1}{\delta T}. \tag{27}$$

The RDMI, RDMD, and RDTD matrices improve the quality of the signal and extract spatial and temporal differences between virtual channels, while simultaneously reducing the dimensionality of the feature space.

I/Q derivative vector: The I/Q derivative vector describes how each complex value in the I/Q vector is changing over time:

$$\text{iq}'^{(T)}_i = \frac{\text{iq}_i^{(T)} - \text{iq}_i^{(T-1)}}{\delta T}. \tag{28}$$

I/Q derivative sum: The I/Q derivative sum scalar describes how the entire I/Q vector is changing over time:

$$\hat{\text{iq}}'^{(T)} = \sum_{i=1}^{|\text{iq}|} \text{iq}'^{(T)}_i. \tag{29}$$

I/Q maximum channel angle: The I/Q max channel angle describes the angle between the complex value in each channel with the maximum magnitude:

$$\bar{\text{iq}}^{(T)} = \arccos\left(\frac{\alpha\beta}{|\alpha||\beta|}\right), \tag{30}$$

where α and β are the complex values with the maximum magnitude in the first and second virtual channels.

In combination, these I/Q features are particularly useful for detecting micro gestures, as even the smallest sub-millimeter movement of the hand or fingers results in a detectable change in I/Q.

The Soli spectrogram contains information on how much of the signal is reflected back to each antenna, which frequencies are attenuated, and which frequencies are resonating in a specific frame. Like the range-Doppler, most information in the spectrogram is located within an ROI that correlates with the range bin of the main target.

Spectrogram Multichannel Integration (SPMI): The SPMI matrix combines spectrogram data from the ROI of K virtual channels:

$$\mathbf{SPMI}_{ij}^{(T)} = \frac{1}{M_{ds} N_{ds} K} \sum_{m=1}^{M_{ds}} \sum_{n=1}^{N_{ds}} \sum_{k=1}^{K} SP_{rc}^{(T,k)}. \tag{31}$$

4.7 3D spatial hand tracking

The novel Soli sensing paradigms described above enable fine finger and gesture tracking in constrained mobile applications. We can easily augment these capabilities with traditional radar spatial positioning to track the hand as a whole without significantly increasing hardware or computational complexity. The Soli sensor uses a 2x2 element receive antenna array and two switched transmitters that allow 2D digital beamforming and 3D localization. Discussion of large scale hand tracking is beyond the scope of the current paper.

4.8 Soli Processing Pipeline and HAL

The Soli principles discussed above define a Soli Processing Pipeline (SPP) summarized in Figure 7. In the *signal preprocessing stage*, hardware-dependent analog and digital preprocessing demodulates the raw received signal and forms the point target response $h(t)$. The specific preprocessing steps, implementation,

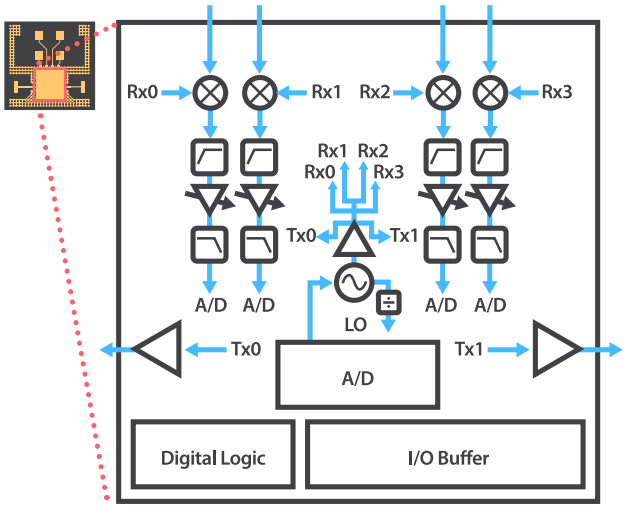


Figure 8: *The Soli hardware architecture features a highly integrated chip with antennas-in-package, RF front end, and baseband processing.*

and point target response form differ by radar modulation schemes, transmission parameters, and hardware architectures. Nevertheless, the representation of the output $s_{rec}(t, T)$ *is the same* (Equations 7-8): though the specific wavelength λ and form of $h(t)$ affect performance metrics such as resolution and accuracy, the rest of the SPP is functionally agnostic to radar modulation, hardware, and transmission parameters.

After the preprocessing stage, a set of signal representations, i.e. *transformations*, are computed using algorithms and techniques described for fast and slow time measurements earlier in the paper. We currently compute I/Q, range-Doppler, range profile, micro-Doppler, fast time spectrogram and three-dimensional spatial profile. These representations provide high-level intuitive insight into the hand's radar response and are *agnostic to specific hardware and modulation schemes*. In other words, any application or algorithm developed on top of these transformations will theoretically work on any radar. Thus these transformations constitute a radar HAL.

Once the transformations are computed, a set of low-dimensional *features* are extracted from the transformation data. The features include, but are not limited to, fine displacement, total measured energy, measured energy from moving scattering centers, scattering center range, velocity centroid, and many others. The design of features that enable robust gesture tracking is a combination of art and science, requiring creative analysis of the data, as well as domain knowledge about the radar and gestures to be captured. Due to the novelty of the current work, we are not aware of any prior art in selection of these features.

Finally, in the last step of the Soli PP, the features are submitted to various machine learning classifiers that are used to identify the exact gesture the user is performing, as well as continuous parameters that define this gesture. Note that the Soli PP does not suggest recovering hand structure or identifying individual fingers. The gesture identification is based on *radar gesture signatures* that are observed in radar transformation space and described through features computed to reflect these gestures. We review gesture recognition algorithms in Section 7 of this paper.

5 Soli Chip and System Architecture

The Soli system architecture is presented in Figure 8. Through the course of our hardware development, we iterated through several implementations of Soli hardware architecture, progressively shrinking the system from a desktop-sized radar box down to a single chip. Our early FMCW radar prototype (Figure 9, top left) was a custom 57-64 GHz radar built out of discrete components using Infineon's BGT60 backhaul communication IC with multiple narrow-beam horn antennas. In parallel, we developed ultra-wideband (UWB) 3-10 GHz impulse radar prototypes based on Novelda's XeThru NVA620x IC [Novelda] (Figure 9, top right), including the design of multiple incorporated coherent receive antennas, such as a hybrid Archimedean Power Spiral.

It quickly became apparent that the form factors and power requirements of these prototypes could not support our ultimate vision of the radar gesture sensor integrated into mobile and wearable devices. Furthermore, in the case of the UWB radar, the centimeter-scale wavelength did not allow for sufficient phase sensitivity required for fine gesture interaction, and the size of the antennas could not be reduced any further while maintaining wide bandwidth and sufficient gain.

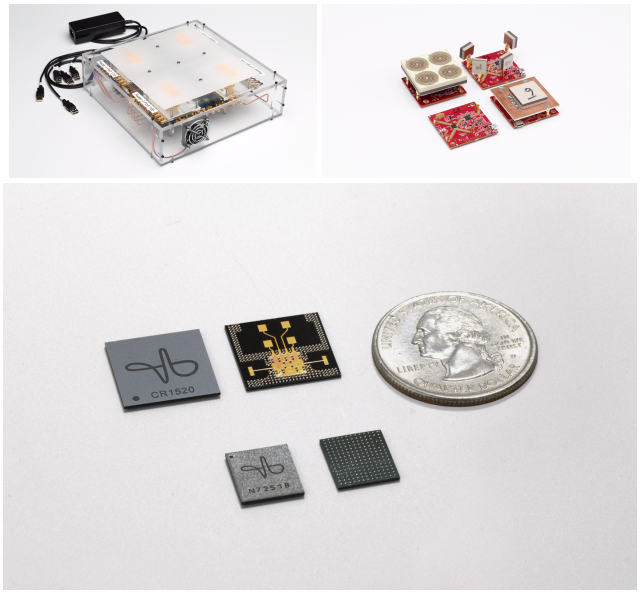


Figure 9: *Top: Early prototypes of Soli FMCW and impulse radars. Bottom: Soli radar 60 GHz chips with antennas-in-package (AiP).*

Recent advancements in semiconductor technology made it possible for us to dramatically miniaturize the radar hardware from our initial prototypes. The increased availability of new CMOS, BiCMOS, and SiGe process technologies with ever-shrinking technology nodes enables high-frequency RF circuitry (>30 GHz) necessary for millimeter-wave radar. Furthermore, modern silicon die packaging technologies allow the placement of antennas directly on the package, either by stacking them vertically on top of the silicon die or by placing then horizontally on the same plane. These technologies allow for complete integration of the sensing radar, including antennas and computation, into a single chip.

In addition to physical miniaturization, the complete integration of radar electronics onto a single semiconductor device allowed us to significantly improve power efficiency and reduce cost. The single chip solution drastically simplifies end-system design by removing the need for GHz-frequency RF development, e.g. antenna and

waveguide design, and expensive materials such as Rogers PCBs with tuned RF properties. In essence, radar becomes a component that can be easily integrated into consumer devices, much like accelerometers, touch sensors and CMOS cameras.

5.1 Soli radar chip

Based on performance evaluation of our early radar prototypes, we developed two fully integrated Soli radar gesture chips, shown at the bottom of Figure 9:

- A 12×12 mm, FMCW SiGe radar chip manufactured by Infineon using embedded Wafer Level Ball Grid Array (eWLB) technology [Brunnbauer et al. 2008].
- A 9×9 mm, direct-sequence spread spectrum (DSSS) CMOS chip manufactured by SiBeam with on-die antennas.

The different process technology used in manufacturing the two chips (SiGe versus CMOS) results in different signal quality, power consumption, and integration capabilities. For example, CMOS allows integration of standard digital logic blocks, mixed-signal blocks and radar technology on the same die, while SiGe does not. The SiGe chip is based on an FMCW modulation scheme common in traditional tracking and imaging radars; the CMOS chip utilizes binary phase-coded DSSS modulation used in communication. Despite these differences, the Soli radar HAL (Section 4.8) allows us to use both chips interchangeably. Detailed discussion of the Soli radar chip is beyond the scope of this paper; see [Nasr et al. 2015; Nasr et al. 2016] for more details of the FMCW radar chip design. Below we highlight the main hardware design principles and features critical for gesture sensing applications:

High level of integration of radar components. Soli radar chips are drop-in sensors that can be integrated into consumer electronic devices with minimal effort. Antennas, RF front end, baseband processing, VCOs and serial communication to program the chip are all integrated on the chip (Figure 8). The CMOS die also includes on-board A/D and digital interfaces. In the future, we plan to further increase the level of component integration.

V-band operation. Both ICs operate at the 60 GHz band, which allows for small antennas, 7 GHz bandwidth utilization, and potential integration with 802.11ad and Wi-Gig standard [Hansen 2011] with minor hardware modifications.

Broad beam antenna-in-package design. Both chips have antenna-in-package (AiP) patch array designed to form a broad 150-degree beam, which allows illumination of hand and fingers at close distance.

Multichannel 2Tx-4Rx for beamforming. The two transmit and four receive antenna elements allow digital beamforming for three-dimensional tracking and coarse spatial imaging. The receive antennas are arranged in a 2×2 pattern optimized for 60 GHz central frequency, enabling classic phased-array beamforming with minimal grating lobes. In both chips we implement beamforming at the receive end. For the SiGe FMCW IC, the beamforming is fully digital without analog phase shifters that are used on CMOS chip.

Low noise and low power. The chip design is optimized for low 1/f noise, phase noise and jitter, which is critical for precise tracking of fine gestures. The power consumption of the radar chips is currently 300 mW with adaptive duty cycling that shuts down the chip during computation. Average power consumption can be further reduced by using application-level duty cycling. We plan to continue optimizing the Soli chip design, e.g. switching to BiCMOS, in order to further decrease power consumption.

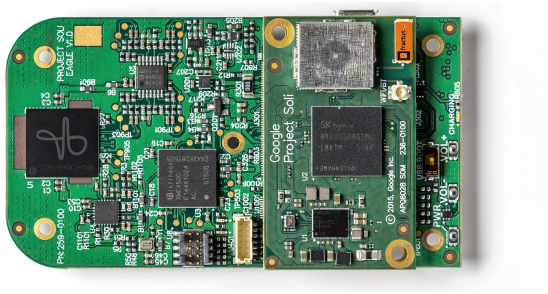


Figure 10: *The Soli chip is easily integrated into development boards as shown here, as well as consumer devices.*

5.2 Soli system design

The Soli chips are designed to be easily integrated into a wide variety of consumer electronics devices. Figure 10 shows an example development board based on the Soli FMCW chip.

The architecture for acquiring and transferring raw radar data varies between the Soli chips. The FMCW chip provides an analogue output and requires an external ADC. Therefore, for data acquisition we used an Infineon XMC4500 Cortex M4 microprocessor with quad 12-bit ADCs running at 1.79 Msample/sec. The acquired data is then streamed over the USB interface to the application processor, e.g. a standard PC or embedded processor executing the Soli Processing Pipeline. In the case of the Soli DSSS radar, data conversion occurs directly on the chip and is then streamed to the application processor via an SPI bus. For both chips, an SPI bus and microprocessor are required to configure and control chip parameters.

Soli chip operation was tested with multiple embedded microprocessors, e.g. 500 MHz dual core Intel Atom and 1.6 GHz Quad-A7 on Snapdragon 400, as well as desktop PCs, including Mac and Linux computers. The performance of the signal processing varies between microprocessors due to differences in architecture and signal processing libraries. Using a standardized range-Doppler transformation test, we measured performance ranging from 1830 transformations per second for a 500 MHz dual-core Intel Edison, to 1200 for a 900 MHz quad-core ARMv7 Cortex-A7, to 49000 for a Linux PC with a 3.4 GHz quad-core Intel i7-4770 microprocessor.

5.3 Soli radar performance evaluation

We validated Soli's fundamental tracking precision by comparing Soli radar displacement measurements with a laser range finder. A metallic plate was mounted perpendicularly to the radar line of sight at an initial position of 50 centimeters from the colocated radar and laser range finder. A robotic arm then moved the plate toward the sensors at varying velocities as displacement data was collected. The Soli radar was operated at RRF = 1 kHz using a single transmitter and single receiver. The radar data was first calibrated using standard distance measurements using the laser range finder.

Figure 11 (a) overlays the displacement trajectories measured by both sensors, while the RMS error between Soli and the laser is shown as a function of velocity in Figure 11 (b). Across twelve different velocities ranging from 20 mm/sec to 200 mm/sec, the average RMS displacement error is 0.43 mm. This number validates Soli's theoretical sub-mm displacement accuracy and indeed approaches the limit of 0.3 mm for an ideal point target, as determined by the Soli radar's phase jitter specification.

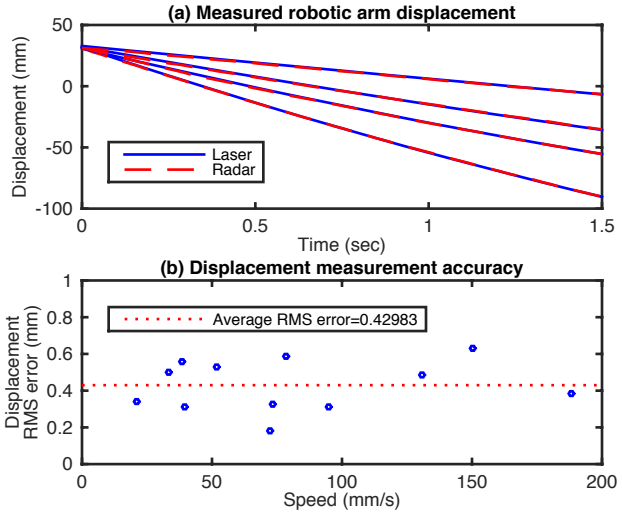


Figure 11: *(a) Comparison of robotic arm displacement measurements for four velocities: 21.146 mm/s, 38.401 mm/s, 51.472 mm/s, and 73.053 mm/s. (b) The average RMS error between the two sensors over twelve different velocities validates Soli's theoretical sub-mm displacement accuracy.*

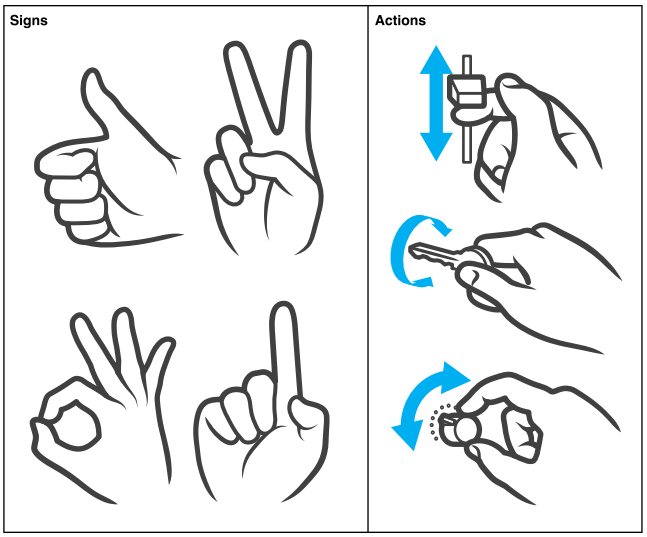


Figure 12: *Sign Gestures (left) describe static hand shapes, while Action Gestures (right) relate to the use of tools and devices.*

6 Soli Interaction Design

Gesture interactions for Soli must build on the strengths of the radar sensor while recognizing human ergonomic and cognitive needs. We found that technical qualities and human needs overlap in a design space we call *micro gestures*: hand-scale finger gestures performed in close proximity to the sensor. These micro gestures exploit the speed and precision of the sensor and avoid the fatigue and social awkwardness associated with the arm- and body-scale gesture interactions enabled by popular camera-based gesture tracking techniques [Khoshelham and Elberink 2012].

6.1 Action Gestures

Gesture recognition with camera-based systems typically interprets image and depth data to build and track a skeletal model of the human body [Shotton et al. 2013] or hand [Weichert et al. 2013]. Soli does not lend itself to this approach because our sensing paradigm proposes tracking in temporal rather then spatial domain. This makes it difficult, though not impossible, to reconstruct object's spatial configurations, such as body postures or static hand shapes, e.g. an open hand or a fist.

Instead of static shapes, the key to Soli interaction is *motion, range* and *velocity*, where the sensor can accurately detect and track components of complex motions caused by a user hand moving and gesturing within sensing field. We therefore focus on using gestures with a clear motion component, that we refer to as *Action Gestures*, rather than gestures that are expressed as static hand shapes, or *Sign Gestures* (Figure 12). As a side note, this quality of the Soli radar sensor also alleviates the privacy concerns of using networked video cameras to track user interactions.

This focus on Action Gestures leads to gestures that are easy to perform, but difficult to interpret and describe, such as the gesture that comprises an action of *the fingertips of thumb and index finger rubbing against each other*. There are no accepted terms that would allow us to communicate such gestures to the users in clear and unambiguous terms. Our solution to this problem is to relate our design language not to symbolic gestures, but to the use of *familiar physical tools and devices* (Figure 12).

6.2 Virtual Tools

Many small, hand-scale physical tools and control mechanisms share qualities that speak to the strengths of both radar sensor and user. Indeed, their operation involves physical motion, requiring users to perform coordinated finger motions at hand-scale. The associated motions are memorable and familiar, easily understood across cultures and can be easily described by referring to the operated tool. At the same time they work at a small scale that will not tire the user and that speak to the strengths of human motor control system that allows very precise, fast and effortless manipulations when small muscles of the hand and fingers are primary actors (see [Zhai et al. 1996]).

The theme of *Virtual Tools* is central to Soli gesture language design, leading to gestures such as a *Virtual Button*, i.e. pressing index finger and a thumb (Figure 15(a)), a *Virtual Slider*, i.e. moving a thumb along the side of the index finger of the same hand (Figure 15(b)) and a *Virtual Dial*, an invisible dial operated between thumb and index finger (Figure 13).

The Soli sensor is capable of recognizing more traditional types of gestures, such as swipes (Figure 15(c,d)). We are evaluating these gestures and believe that they are complementary to the Virtual Tool theme that has been instrumental in guiding interaction design and defining the unique Soli gesture language, which has qualities not easily available in other gesture interfaces. We describe some of these qualities in the rest of this section.

6.2.1 Haptic Feedback

Because the hand both embodies and acts on each Virtual Tool, haptic feedback is generated by the the hand acting on itself. Therefore, unlike other touchless gesture interaction system, e.g. camera-based systems, Soli interaction does not suffer from the lack of haptic feedback, which is an important component of physical controls. Soli interactions can feel tactile even when no actual physical devices are present and no active haptic feedback is generated by the system.

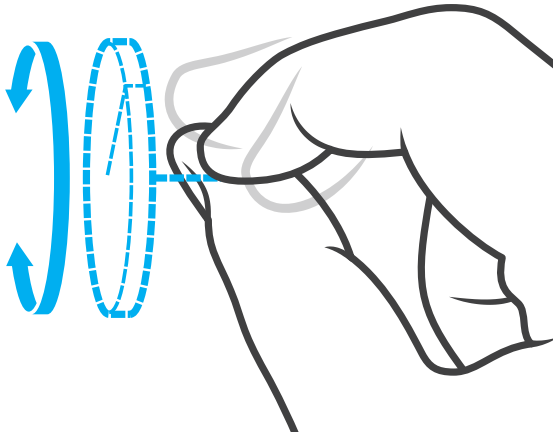


Figure 13: *Virtual Dial Tool*

6.2.2 Proprioception

Virtual Tool interactions rely on the hand and fingers moving in relation to themselves. In this scenario, gesture interaction benefits from *proprioception* - the sense of one's own body's relative position in space and parts of the body relative to each other. When operating a Virtual Slider between index finger and thumb, for example, interactions can be accurately controlled without relying on hand-eye coordination. As long as the hand's position in relation to the sensor is established, a user does not need to look at the interface they are controlling; the feedback from his or her own hand should be sufficient.

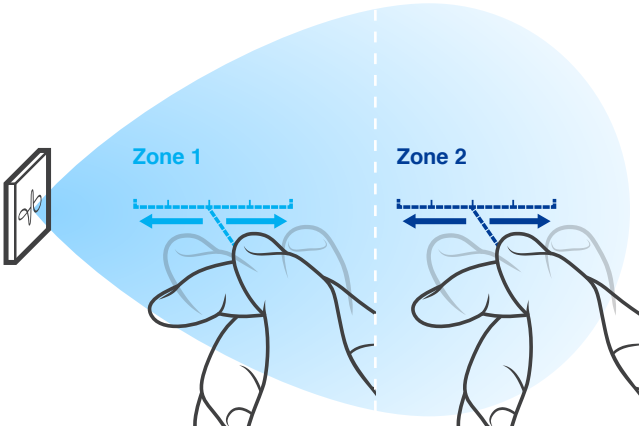


Figure 14: *Virtual Slider Tool in two zones*

6.2.3 Interaction Zones

Because radar can track range independently of motion, most Virtual Tools can be mapped to different interface features depending on where they are performed in relation to the sensor, a concept that is sometimes referred to as spatial multiplexing. For example, a Virtual Button gesture performed close to the sensor may cause a map to zoom in, while the same gesture performed further away from the sensor may cause the map to zoom out (Figure 14). The combination of Virtual Tools and Interaction Zones makes it possible to design interfaces where a complex set of features can be controlled by a minimal set of gestures.

6.3 Feedback

Visual or audio feedback is a crucial component of any user interface, and even more so for Soli because interactions happen in-air, in an invisible field. This feedback can be thought of as a *rollover* effect in desktop GUIs, but mapped to the unique states of the Soli interactive system. Among others, these states include the concepts of *presence*, i.e. indicating when a hand is inside the sensing field, *proximity* or spatial multiplexing described above, and *activity*, such as when a Virtual Dial interaction starts and stops.

6.4 Virtual Toolkits

Due to the nature of the Soli gesture recognition pipeline (Section 7.1), Soli interactions are defined as a finite set of gestures. It is generally the case that the reliability of gesture recognition decreases as the number of gestures in a given set increases. This characteristic of gesture recognition is matched by a human requirement to minimize the number of new gestures and interactions that must be learned in order to operate a system. Building on the idea of Virtual Tools, we organize Soli interactions into *Virtual Toolkits*: small sets of gestures which, in combination, are easy to detect, easy for users to remember and sufficiently complex to support the control of everyday user interfaces.

7 Gesture Recognition with Soli

The fundamental approach to gesture recognition with Soli is to exploit the temporal accuracy of radar. Therefore, we recognize gestures directly from temporal variations in the received radar signal by extracting and recognizing *motion signatures* in the Soli transformations. This is contrary to many existing gesture sensing approaches that are primarily spatial and explicitly estimate a hand pose or skeletal model prior to recognizing gestures [Romero et al. 2013; Keskin et al. 2013]. In addition to exploiting the temporal accuracy of radar, Soli's gesture recognition was designed to (i) maintain the high throughput of the sensor to minimize latency; (ii) exploit advantages of using multiple antennas to maximize SNR and improve recognition accuracy; (iii) provide both discrete and continuous predictions[3]; (iv) be computationally efficient to work on miniature, low-power devices, e.g. smart watches.

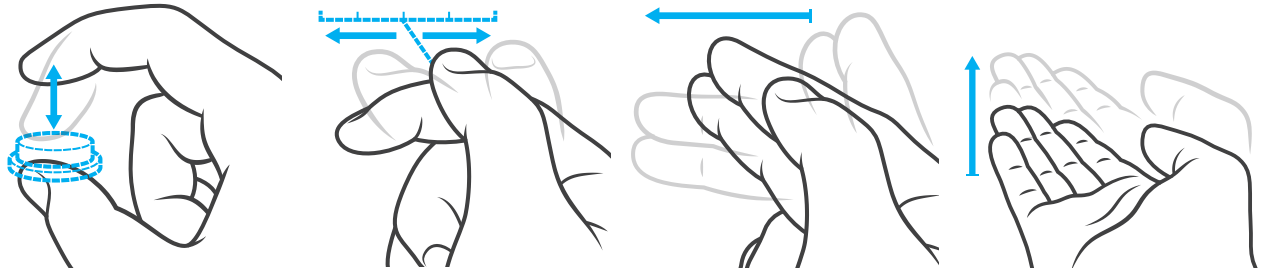
Meeting these design goals is far from trivial and requires various trade-offs, such as the balance between recognition accuracy, detection latency, and computational efficiency. In this section, we describe these challenges and present the basic building blocks of the Soli gesture recognition pipeline that can be used to build a variety of radar-based gesture systems.

To evaluate our approach, we implemented one possible gesture recognition pipeline based on classic Random Forest classifier [Breiman 2001] for a basic gesture set. This exemplar pipeline demonstrates one of many possible gesture pipeline implementations and validates the premise that Soli can be used for computationally efficient, real-time gesture recognition that can potentially run on devices with low computational power, e.g. wearables.

7.1 Soli Gesture Recognition Pipeline

Soli's exemplary gesture recognition pipeline consists of the following three blocks:

[3] An example of a discrete Soli gesture is the event triggered by a virtual button. Alternatively, an example of a continuous Soli gesture is the continuous value output by a virtual slider movement.



(a) *Virtual Button.* **(b)** *Virtual Slider.* **(c)** *Horizontal Swipe* **(d)** *Vertical Swipe*

Figure 15: *The four gestures recognized in the gesture evaluation.*

1. *Feature Extraction:* Low-dimensional and gesture-specific features are computed from the transformations;
2. *Gesture Inference:* Gesture recognition is performed using appropriate machine learning classifiers; and
3. *Filtering:* Temporal and contextual filtering is performed to improve the quality of recognition.

We discuss these three blocks of the Soli gesture recognition pipeline in the rest of this section.

7.1.1 Feature Extraction

Like many machine learning tasks, extracting salient features from the radar signals is a critical step for building robust gesture recognition pipeline [Guyon and Elisseeff 2003]. This is particularly important for recognition of temporal gestures, where temporal features provide information on gesture evolution over time. It is important to emphasize that the choice of features is entirely application and gesture dependent: although some generic features, e.g. target range, velocity of scattering centers or velocity centroid discussed earlier in the paper (Section 4.5) can be effective across multiple gestures, they are usually not sufficient. Narrowly tuned gesture-specific features are almost always necessary to achieve robust gesture recognition. Due to absence of prior art on gesture recognition with mm-wave radars, *feature discovery* is an important part of the development process.

There is no single approach for feature discovery. In our work, we combine signal observation and intuition with domain knowledge about radar operation and constraints imposed by interaction. In addition we heavily employ a *machine learning perspective*, where the most relevant features are automatically detected and selected during the learning phase. For example, when using a Random Forest algorithm, the features selected by the classifier during the learning phase can be analyzed to provide a feature importance for a specific gesture set. Similar techniques are available for other machine learning algorithms [Storcheus et al. 2015; Bishop 2006].

In Soli we focus on *temporal, dynamic hand gestures*, as opposed to static hand postures. This makes it challenging to recognize a gesture by using features from a single sample of radar data taken at time T. To mitigate this issue, some of the features in Soli gesture recognition pipeline are extracted over a sliding temporal window. Here, features are collected in a temporal buffer and with each new sample the contents of the buffer is concatenated into a feature vector. Furthermore, *meta features* can be computed from the contents of the temporal buffer, such as basic statistics of feature evolution over time, including the mean, standard deviation, root mean squared deviation, frequency and others. The resulting features extracted from the temporal window by both approaches are then combined to build a feature vector that forms the input to a machine learning algorithm for gesture classification.

7.1.2 Gesture Inference

There are a number of powerful classification algorithms that can be used for temporal gesture recognition, such as Hidden Markov Models [LoPresti et al. 2015; Kurakin et al. 2012], Dynamic Time Warping [Wu et al. 2013; Gillian et al. 2011], Support Vector Machines [Dardas and Georganas 2011; Gillian 2011], or Convolutional Neural Networks [Ji et al. 2013; Duffner et al. 2014; Molchanov et al. 2015b]. These algorithms are computationally expensive and are not suited for real time operation on low-power embedded platforms at high frame rates and small memory footprint. By benchmarking and comparing various algorithms we converged on a *Random Forest* classifier.

Random Forests have proven fast and effective multi-class classifiers [Shotton et al. 2013; Song et al. 2014] and found preferable for Soli gesture recognition pipeline due its computational speed, low memory footprint, and generalization ability. Indeed, in our evaluations in Section 7.2.2, Soli gesture recognition pipeline runs at up to 2080 classifications per second on Qualcomm Snapdragon application processor that is commonly used in smart watches. Furthermore, the size of the model produced by the Random Forest is a small percentage of that of other classifiers, such as Support Vector Machines, which produced equivalent classification results in our prior classification benchmarking.

The small model footprint and efficient real-time classification are important factors for choosing Random Forests for Soli pipeline. In scenarios where memory, power and computation are not restricted, other powerful classification approaches can be also applied.

7.1.3 Filtering

To *improve the accuracy of the predictions* made by the classifier, the raw predictions are improved using a *Bayesian filter*. We take advantage of the high frame rate of our pipeline and the temporal correlation in neighboring predictions. Indeed, even fast gestures occur over dozens or even hundreds of frames. We exploit this temporal coherency to filter the raw predictions made by the classifier using a Bayesian filter, which significantly reduces sporadic false-positive gesture errors, while maintaining a minimal prediction latency for the end application (Figure 16). In fact, with temporal filtering, some of the temporal gestures can be recognized *before* the gesture has been completed. Figure 16 for instance shows how the classification probabilities for a swipe gesture start to rapidly increase at the very start of the gesture, peaking at the end.

Unlike other temporal filters that simply average predictions over time, the Soli Bayesian gesture filter takes into account two impor-

tant *priors*: (1) a *temporal prior* based on a weighted average of the most recent predictions; (2) a *contextual prior*, consisting of information passed to the Soli library from the end application listening for gesture events.

The Bayesian filtered posterior probability for gesture k at time T with feature vector $\mathbf{x}$ is given by

$$P(g_k|\mathbf{x}) = \frac{P(\mathbf{x}|g_k)P(g_k)}{\sum_j P(\mathbf{x}|g_j)P(g_j)}, \tag{32}$$

where the likelihood $P(\mathbf{x}|g_k)$ is the raw prediction likelihood output by the classifier at time T. The prior $P(g_k)$ consists of the temporal prior, combined through a weighted average over the previous N predictions from the classifier, and contextual prior z_g. The contextual prior is passed from the end application to the Soli library to indicate how likely (or unlikely) a specific gesture might be in the current state of the application at time T:

$$P(g_k^{(T)}) = z_k^{(T)} \sum_{n=1}^{N} w_n P(\mathbf{x}^{(T-n)}|g_k^{(T-n)})^{(T-n)}, \tag{33}$$

where w_n is the filter weight, set to increase weight of more recent predictions. The filtered prediction at time T is given by

$$g^* = \arg\max_k \quad P(g_k|\mathbf{x}), \quad 1 \leq k \leq K. \tag{34}$$

7.1.4 Gesture Spotting & Temporal Variation

When we combine the feature extraction, gesture inference and temporal filtering, the result is the general structure of a Soli gesture recognition pipeline. To apply this pipeline to real-time gesture recognition on a continuous stream of unsegmented data, the following two challenges must be addressed: gesture spotting and temporal gesture variation.

To address *gesture spotting* for Soli, we explicitly provide both positive and negative examples (i.e., valid gestures and background movements) to the machine learning algorithms for training. In addition, we use a continuous stream of time series data as input to the classification pipeline, combined with the Bayesian filter to smooth predictions over time and remove sporadic false positive detection errors. This is further improved with design constraints placed on the system, as described in Section 6.

To account for *temporal gesture variation* we provide a large number of gesture examples performed by multiple different users. Using this data, we train our machine learning model to be as invariant to the gesture as possible.

7.2 Recognizing Button, Slider and Swipe Gestures

In this section, we evaluate a Soli gesture recognition pipeline on the following four gestures: *Virtual Button*, *Virtual Slider*, *Horizontal Swipes* and *Vertical Swipes* (see Figure 15). A *Virtual Button* is a double-tap gesture between the thumb and index finger. A *Virtual Slider* is a continuous gesture that consists of the thumb 'sliding' horizontally along the index finger to create a continuous controller. This is combined with a 'select' gesture consisting of the thumb tapping the index finger in a vertical direction. Finally, *Swipes* are hand swipes performed in two directions above the sensor: 1) swiping horizontally from right-to-left across the sensor; and 2) swiping vertically up-and-away from the sensor.

7.2.1 Configuring Gesture Recognition Pipeline

We configured a Soli sensor to sample raw radar frames at 2500 frames per second using two virtual channels. We elected to use two virtual channels to improve the SNR and extract inter-channel features. Transformations were computed for both channels every 10 frames, resulting in 250 transformations per second. The raw sample rate was set as high as possible to capture precise velocity information for the virtual tool gestures and to mitigate velocity aliasing for the larger swipe gestures. While the sensor can capture data at significantly higher rates, our sample rate was constrained by the radar frame size and the number of active channels. Additionally, instead of computing the Soli transformations on a 1:1 ratio for each raw radar frame, we computed them on a 1:10 ratio. This significantly reduced the computational load, while maintaining a high sample rate to mitigate velocity aliasing. The Soli software library (Section 8) enables a developer to easily tune these parameters to balance the responsiveness of the system with performance and efficiency.

To recognize four test gestures and the background class we used nine low-dimensional features: range, acceleration, velocity, velocity centroid, total energy, moving energy, movement index, fine displacement and target detection (Section 4.8). In addition we computed the RDMI, RDMD, and RDTD matrices from the range-Doppler ROI that was set to 20% of the original height and 100% width. This resulted in matrices sizes of 6x32 which were further down-sampled to the size of 3x8. In addition to the range-Doppler-centric features, we used the I/Q derivative sum and I/Q maximum channel angle features.

The features above all corresponded to one frame of transformation data at time T. To account for the temporal nature of the gestures, the features from each frame were buffered into a sliding window. The size of the temporal window was set to 10 frames, which covered approximately 40 milliseconds. For each of the low-dimensional tracking features, we computed five meta-features: mean, standard deviation, root mean squared, integrated sum and integrated delta. At each new update to the temporal buffer, the features above were computed and concatenated into one main *feature vector* used as input to the Random Forest classifier. The resulted feature vector included 785 features: 3x8x3x10 RD ROI features, 2x10 I/Q features and 9x5 tracking meta features.

An advantage of combining the data-centric features, tracking features, RF energy features, and meta features in one feature vector was that the Random Forest classifier automatically detected and selected relevant features during the learning phase. This allowed us to disable the computation of irrelevant features and improve speed of classification.

7.2.2 Gesture Recognition Evaluation

We recorded five users performing 50 repetitions of the four gestures shown in Figure 15 at various locations within a 30 cm range of the sensor. This was in addition to generic background movements performed over the sensor and natural transitions in and out of valid gestures. Participants were seated in front of a stationary sensor, instructed in how to perform each gesture, and given several minutes to practice until they were comfortable with each gesture. Each participant performed the 50 repetitions of each gesture *twice*, with a break in between the two sessions. The first time series recording for each user was grouped into a single data set for training the evaluation model. The second time series recording for each user was grouped into an independent data set for testing the evaluation model. We opted for an independent test data set over leave-N-out, cross validation, or other random sampling

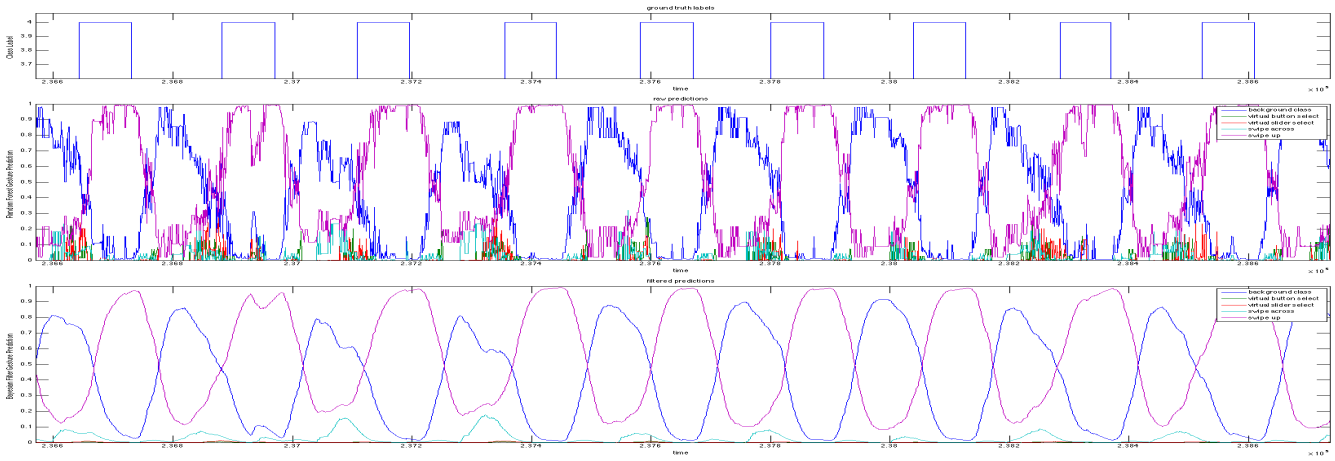


Figure 16: *Ground truth labels (top), raw likelihoods from the random forest classifier (middle), and filtered predictions from the Bayesian filter (bottom) for a small subset of the test data set containing nine vertical swipe gestures. The square wave in the ground truth labels indicates sections in the time series containing valid swipe gestures.*

	per-sample accuracy	per-gesture accuracy
raw	73.64%	86.90%
filtered	78.22%	**92.10%**

Table 1: *The accuracy for the per-sample and per-gesture results for 1000 test gestures (308,335 test samples) for five users. Raw indicates the direct estimation of the Random Forest classifier, filtered indicates the estimation of the Bayesian filter.*

techniques, as we noted unrealistically high results with these validation techniques that did not translate to real-time generalization. This was due to the strong correlation in samples because of the high frame rates of the sensor and the time series nature of the data.

In total, the training and test data sets resulted in 2000 gesture examples (5 participants x 4 gestures x 50 repetitions x 2 sessions) or approximately 6 million raw data frames giving 600,000 Soli transformations. A Random Forest classifier was trained using a forest size of 50 with a maximum depth of 10. The trained model was then tested using the 1000 test gestures. We used the following metrics: 1) the accuracy of the system *per-sample*; and 2) the accuracy *per-gesture*. They were further evaluated for the *raw prediction*, i.e. the direct output of the Random Forest classifier, and the *filtered prediction*, i.e. the output of the Bayesian filter.

The results are presented in Table 1 and demonstrate a filtered per-gesture accuracy of 92.10% over the 1000 test gestures. Figure 16 shows the raw and filtered predictions for a small subset of the test data set. This illustrates a significant improvement between the raw and filtered predictions for both the per-sample and per-gesture analysis. This directly translates to real-time prediction accuracy, as the Bayesian filter significantly reduces the number of false-positive gesture errors. Figure 16 also explains a majority of the 7.9% error in the test data set, as there was a number of samples at the start and/or end of a gesture that were classified incorrectly as the background class, reducing the overall accuracy of the prediction. In Section 8, we show that this entire pipeline can be run in real-time on embedded platforms using the Soli Software Pipeline.

8 Performance of Soli Software Pipeline

The Soli Software Pipeline (SSP) supports real-time gesture recognition using multiple radar hardware architectures and can be optimized for embedded application processors to allow high frame rates with minimized latency for improved temporal resolution. The SSP can be modularized into the following main stages:

1. Capturing raw radar data from a Soli sensor or from a pre-recorded file.
2. Processing the raw radar data using Soli DSP algorithms outlined in Section 4.8 and computing transformations. The resulting DSP transformations are hardware agnostic at the output of the Soli HAL.
3. Computing custom features from the Soli transformations, such as tracking the position of a user's hand.
4. Recognizing gestures using machine learning classifiers.
5. Triggering callbacks to pass data and notifications to the main application.

The entire SSP was tested and benchmarked on various embedded platforms such as the Raspberry Pi2 running at 900 MHz and the Qualcomm Snapdragon 400 (APQ8028) running at 1.6 GHz. A test program utilized the Soli library to process 10 seconds of captured data for one channel, 64 samples per channel, at a total of 8422 frames. Throttling was turned off to measure the maximum frame rate, running one thread at 100% CPU utilization.

We measured the SSP performance for two test implementations: 1) fine displacement computation as described in Equation 19, and 2) full gesture recognition pipeline as described in Section 7.2.2. The pipelines were tested on the following system configurations: Raspberry Pi2 with no NEON optimized DSP functions; Raspberry Pi2 with NEON optimized DSP functions and the FFTS open source FFT library optimized for ARM SIMD; and Qualcomm Snapdragon with NEON optimized DSP functions and the FFTS library. The ARM versions were built using the GCC 4.9 open source compiler.

The results of the performance evaluation demonstrated that with no optimizations, the Soli fine displacement pipeline runs on the order of 10,000 fps unthrottled on the Raspberry Pi2. With embedded optimizations, the Soli gesture recognition pipeline can run at greater than 2,800 fps on the Qualcomm Snapdragon 400. These large frame rates ensure temporal resolution and minimized latency

Platform	Fine Displacement	GR
RPi2 (no NEON-opt)	8000 fps	669 fps
RPi2 (NEON-opt)	11500 fps	1480 fps
Snapdragon (NEON-opt)	18000 fps	2880 fps

Table 2: *Performance of SSPs on embedded platforms for fine displacement computation ("Fine Displacement") and the gesture recognition pipeline outlined in Section 7.2.2 ("GR"). The maximum frames per second (fps) is measured while consuming one thread at 100% CPU utilization.*

to enable fine gesture control. Furthermore, these optimizations enable the Soli pipeline to run at lower frame rates when applicable and consume a small percentage of a single CPU resources.

Overall, this performance evaluation demonstrates that the SSP is lightweight and its efficient implementation enables touchless gesture interaction on low-power embedded platforms used in wearable, mobile and IoT applications.

9 Applications and Future Work

Soli proposes a new category of gesture sensors based on physical principles of millimeter-wave RF radiation that were not previously explored in interactive applications. This paper demonstrated that radars are indeed a viable, powerful and attractive technology that can enhance and improve user interaction. We presented an RF silicon sensor uniquely designed for fine hand gesture interaction, based on new principles of radar hardware design and signal processing. Furthermore, our gesture recognition and software pipelines demonstrated that we can achieve fast and fluid gestures at high update and recognition rates running on light-weight embedded systems such as smart watches.

9.1 Applications of Soli

There are numerous applications of this new technology. Due to the small form factor, low power consumption and low cost of RF-based gesture sensors, we can envision use of this technology in wearable and mobile interaction, smart appliances and Internet of Things, mobile VR and AR systems, interactive robots and drones, game controls, smart objects and garments, massively interactive physical environments, as well as novel techniques for scanning and imaging physical environment, way finding, accessibility and security, mobile spectroscopy, and many other exciting applications.

Among all of these possibilities, we chose to focus on close range sensing of fine and fluid gestures based on the Virtual Tools metaphor that we proposed in this paper. We envision that the same vocabulary of simple touchless gestures can control a broad range of devices across multiple application categories. For example, the same gesture that allows one to navigate a map on a smart watch can be used to control music on a radio, manipulate graphics on a tablet, or simply set an alarm clock. Thus Soli could lead to a truly ubiquitous gesture language that, once mastered, would allow a user to control smart computing devices with one, universal gesture set, without relying on physical controls. The interface would disappear and the user's hand would become the only input device he or she would ever need.

In many ways, the vision behind Soli is similar to the computer mouse, which brought universality of interaction to desktop computing, expanding users' access to technology. It paved the way for the Internet, mobile applications and the current expansion of computing, dramatically increasing our access to information, content and services, bringing new quality and enjoyment to our lives.

9.2 Challenges and Future Work

Soli technology is in a very early stage of development. There is very little prior art available and this paper reports only initial steps in exploring this new sensing modality, leaving open a wide field for exploration and novel research.

A significant amount of future work is required to address many fundamental challenges intrinsic to all radar sensing, including radar clutter, signal coupling, multi-path, fading, interference, and occlusion, among others. The effects of these phenomena on radar gesture sensing are not yet well understood and present several new avenues for research and technical development. Further experimental characterization of the Soli sensor is also needed, along with quantitative understanding of the effect of various radar parameters, e.g. frame rate and bandwidth among many others, on fine gesture tracking and recognition performance.

One of the most important and uncharted directions for future research lies in exploring new machine learning and gesture recognition approaches that can exploit the fundamental strengths of the Soli sensor. Our current work demonstrated the validity of Soli by designing a basic gesture recognition pipeline for a small set of gestures, evaluated for a limited number of test users. We anticipate alternative and improved gesture recognition approaches that allow for robust gesture interaction across larger gesture sets and many users. In particular, we look forward to future work that would fully implement the Virtual Tools paradigm proposed in this paper, extending the initial steps we have taken in this direction.

Yet another important area of future research on Soli is the human factors implications of these new interaction modalities. There are also many exciting opportunities to discover and develop novel interaction techniques, applications and use cases of this technology.

With this paper, we hope to inspire a broad and diverse community of researchers, scientists and practitioners to explore this new and exciting technology. We believe the full potential of Soli and RF as a new sensing paradigm for human computer interaction and interactive graphics of the future has yet to be realized.

Acknowledgements

We thank the Touchstone team at Google ATAP, in particular Roberto Aiello and Changzhan Gu for their technical expertise and assistance, as well as our close collaborators Timo Arnall and Jack Schulze of Ottica; Jonas Loh, Stephan Thiel, and Steffen Fiedler of Studio NAND; Tim Gfrerer; Arturo Castro; Hannah Rosenberg; Derek Yan; and Prof. Howard Zebker of Stanford University. We additionally thank Regina Dugan for her leadership and support at Google ATAP. Finally, we gratefully acknowledge the many contributions of our industry partners at Infineon, especially Saverio Trotta; SiBeam; BluFlux; FlatEarth; CEI; and BDTI.

References

AXELSSON, S. R. 2004. Noise radar using random phase and frequency modulation. *IEEE Transactions on Geoscience and Remote Sensing 42*, 11, 2370–2384.

AZEVEDO, S., AND MCEWAN, T. 1996. Micropower impulse radar. *Science and Technology Review*, 17–29.

BARNES, S. B. 1997. Douglas Carl Engelbart: Developing the underlying concepts for contemporary computing. *IEEE Annals of the History of Computing 19*, 3, 16–26.

BAUM, C. E., ROTHWELL, E. J., CHEN, K.-M., AND NYQUIST, D. P. 1991. The singularity expansion method and its application to target identification. *Proceedings of the IEEE 79*, 10, 1481–1492.

BENEZETH, Y., JODOIN, P. M., EMILE, B., LAURENT, H., AND ROSENBERGER, C. 2008. Review and evaluation of commonly-implemented background subtraction algorithms. In *ICPR 2008*.

BISHOP, C. M. 2006. Pattern recognition. *Machine Learning*.

BOLT, R. A. 1980. Put-that-there. *SIGGRAPH '80*, 262–270.

BREIMAN, L. 2001. Random forests. *Machine Learning 45*, 1, 5–32.

BROOKNER, E. 1985. Phased-array radars. *Scientific American 252*, 2, 94–102.

BROWN, L. 1999. *A Radar History of World War II: Technical and Military Imperatives*. Institute of Physics Publishing.

BRUNNBAUER, M., MEYER, T., OFNER, G., MUELLER, K., AND HAGEN, R. 2008. Embedded wafer level ball grid array (eWLB). In *IEMT 2008*, IEEE, 1–6.

CARD, S., MORAN, T., AND NEWELL, A. 1983. *The Psychology of Human-Computer Interaction*. L. Erlbaum Associates.

CHAN, L., CHEN, C.-H. H. Y.-L., AND YANG, S. 2015. Cyclops: Wearable and single-piece full-body gesture input devices. In *CHI 2015*, ACM, 3001–3009.

COMSCORE INC. 2014. The US mobile app report. Tech. rep.

COOPERSTOCK, J. R., FELS, S. S., BUXTON, W., AND SMITH, K. C. 1997. Reactive environments. *Communications of the ACM 40*, 9 (Sep), 65–73.

DARDAS, N. H., AND GEORGANAS, N. D. 2011. Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques. *IEEE Transactions on Instrumentation and Measurement 60*, 11, 3592–3607.

DIETZ, P., AND LEIGH, D. 2001. Diamond Touch. In *UIST '01*, 219.

DORFMULLER-ULHAAS, K., AND SCHMALSTIEG, D. 2001. Finger tracking for interaction in augmented environments. *ISMAR 2001*, 55–64.

DUFFNER, S., BERLEMONT, S., LEFEBVRE, G., AND GARCIA, C. 2014. 3D gesture classification with convolutional neural networks. In *ICASSP 2014*, IEEE, 5432–5436.

FCC, 2016. FCC online table of frequency allocations.

GENG, S., KIVINEN, J., ZHAO, X., AND VAINIKAINEN, P. 2009. Millimeter-wave propagation channel characterization for short-range wireless communications. *IEEE Transactions on Vehicular Technology 58*, 1 (Jan), 3–13.

GILLIAN, N., KNAPP, R. B., AND O'MODHRAIN, S. 2011. Recognition of multivariate temporal musical gestures using n-dimensional dynamic time warping. In *Proceedings of the 2011 International Conference on New Interfaces for Musical Expression (NIME 11), Oslo, Norway*.

GILLIAN, N. 2011. *Gesture Recognition for Musician Computer Interaction*. PhD thesis, School of Music and Sonic Arts, Queen's University Belfast.

GUSTAFSON, S., HOLZ, C., AND BAUDISCH, P. 2011. Imaginary phone: Learning imaginary interfaces by transferring spatial memory from a familiar device. *UIST '11*, 283–292.

GUYON, I., AND ELISSEEFF, A. 2003. An introduction to variable and feature selection. *The Journal of Machine Learning Research 3*.

HANSEN, C. J. 2011. WiGiG: Multi-gigabit wireless communications in the 60 GHz band. *IEEE Wireless Communications 18*, 6, 6–7.

HARRISON, C., TAN, D., AND MORRIS, D. 2010. Skinput: Appropriating the body as an input surface. *CHI 2010*, 453–462.

HOLLEIS, P., SCHMIDT, A., PAASOVAARA, S., PUIKKONEN, A., AND HÄKKILÄ, J. 2008. Evaluating capacitive touch input on clothes. In *MobileHCI '08*, ACM Press, New York, New York, USA, 81–90.

HOLZ, C., AND WILSON, A. 2011. Data miming: Inferring spatial object descriptions from human gesture. *CHI 2011*, 811–820.

HUSSAIN, M. G. 1998. Ultra-wideband impulse radar – an overview of the principles. *IEEE Aerospace and Electronic Systems Magazine 13*, 9, 9–14.

JI, S., XU, W., YANG, M., AND YU, K. 2013. 3D convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 35*, 1, 221–231.

JUNKER, H., AMFT, O., LUKOWICZ, P., AND TRÖSTER, G. 2008. Gesture spotting with body-worn inertial sensors to detect user activities. *Pattern Recognition 41*, 6, 2010–2024.

KELLER, J. B. 1962. Geometrical theory of diffraction. *JOSA 52*, 2, 116–130.

KELLOGG, B., TALLA, V., AND GOLLAKOTA, S. 2014. Bringing gesture recognition to all devices. In *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14)*, 303–316.

KESKIN, C., KIRAÇ, F., KARA, Y. E., AND AKARUN, L. 2013. Real time hand pose estimation using depth sensors. In *Consumer Depth Cameras for Computer Vision*. Springer, 119–137.

KHOSHELHAM, K., AND ELBERINK, S. O. 2012. Accuracy and resolution of Kinect depth data for indoor mapping applications. *Sensors 12*, 2, 1437–1454.

KIM, D., HILLIGES, O., IZADI, S., BUTLER, A. D., CHEN, J., OIKONOMIDIS, I., AND OLIVIER, P. 2012. Digits. *UIST '12*, 167–176.

KNOTT, E. F. 2012. *Radar Cross Section Measurements*. Springer Science & Business Media.

KURAKIN, A., ZHANG, Z., AND LIU, Z. 2012. A real time system for dynamic hand gesture recognition with a depth sensor. In *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, 1975–1979.

LEE, S., AND BUXTON, W. 1985. A multi-touch three dimensional touch-sensitive tablet. In *CHI'85*, 21–25.

LEVANON, N. 2000. Multifrequency complementary phase-coded radar signal. *IEE Proceedings - Radar, Sonar and Navigation 147*, 6, 276–284.

LOPRESTI, L., LACASCIA, M., SCLAROFF, S., AND CAMPS, O. 2015. Gesture modeling by Hanklet-based hidden Markov model. In *Computer Vision – ACCV 2014*, D. Cremers, I. Reid,

H. Saito, and M.-H. Yang, Eds., vol. 9005 of *Lecture Notes in Computer Science*. Springer International Publishing, 529–546.

MELGAREJO, P., ZHANG, X., RAMANATHAN, P., AND CHU, D. 2014. Leveraging directional antenna capabilities for fine-grained gesture recognition. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ACM, 541–551.

MOLCHANOV, P., GUPTA, S., KIM, K., AND KAUTZ, J. 2015. Hand gesture recognition with 3D convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 1–7.

MOLCHANOV, P., GUPTA, S., KIM, K., AND PULLI, K. 2015. Multi-sensor system for driver's hand-gesture recognition. In *IEEE Conference on Automatic Face and Gesture Recognition*.

MOLCHANOV, P., GUPTA, S., KIM, K., AND PULLI, K. 2015. Short-range FMCW monopulse radar for hand-gesture sensing. In *IEEE Radar Conference (RadarCon 2015)*, IEEE, 1491–1496.

NASR, I., KARAGOZLER, E., POUPYREV, I., AND TROTTA, S. 2015. A highly integrated 60-GHz 6-channel transceiver chip in 0.35 um SiGe technology for smart sensing and short-range communications. In *IEEE CSICS 2015*, IEEE, 1–4.

NASR, I., JUNGMAIER, R., BAHETI, A., NOPPENEY, D., BAL, J. S., WOJNOWSKI, M., KARAGOZLER, E., RAJA, H., LIEN, J., POUPYREV, I., AND TROTTA, S. 2016. A highly integrated 60 GHz 6-channel transceiver with antenna in package for smart sensing and short range communications. *submitted to IEEE Journal of Solid State Circuits*.

NATHANSON, F. E., REILLY, J. P., AND COHEN, M. N. 1991. *Radar Design Principles – Signal Processing and the Environment*, vol. 91.

NOVELDA. Novelda Xethru NVA620x IC.

OTERO, M. 2005. Application of a continuous wave radar for human gait recognition. *Proceedings of SPIE 5809*, 538–548.

PARADISO, J., ABLER, C., HSIAO, K.-Y., AND REYNOLDS, M. 1997. The Magic Carpet: Physical sensing for immersive environments. In *CHI'97 Extended Abstracts on Human Factors in Computing Systems*, ACM, 277–278.

PARADISO, J. A. 1999. The Brain Opera technology: New instruments and gestural sensors for musical interaction and performance. *Journal of New Music Research 28*, 2, 130–149.

PARK, J.-I., AND KIM, K.-T. 2010. A comparative study on ISAR imaging algorithms for radar target identification. *Progress In Electromagnetics Research 108*, 155–175.

POTTER, L. C., CHIANG, D.-M., CARRIERE, R., AND GERRY, M. J. 1995. A GTD-based parametric model for radar scattering. *IEEE Transactions on Antennas and Propagation 43*, 10, 1058–1067.

PU, Q., GUPTA, S., GOLLAKOTA, S., AND PATEL, S. 2013. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking*, ACM, 27–38.

RAHMAN, T., ADAMS, A. T., ZHANG, M., AND CHOUDHURY, T. 2015. DoppleSleep: A contactless unobtrusive sleep sensing system using short-range doppler radar. In *Ubicomp '15*, ACM.

RALSTON, T. S., CHARVAT, G. L., AND PEABODY, J. E. 2010. Real-time through-wall imaging using an ultrawideband MIMO phased array radar system. In *IEEE International Symposium on Phased Array Systems and Technology (ARRAY)*, IEEE.

REKIMOTO, J. 2001. GestureWrist and GesturePad: Unobtrusive wearable interaction devices. *ISWC 2001*, 21–27.

RICHARDS, M. A., SCHEER, J., HOLM, W. A., ET AL. 2010. *Principles of modern radar: basic principles*. SciTech Pub.

ROMERO, J., KJELLSTRÖM, H., EK, C. H., AND KRAGIC, D. 2013. Non-parametric hand pose estimation with object context. *Image and Vision Computing 31*, 8, 555–564.

RUSSELL, D., STREITZ, N., AND WINOGRAD, T. 2005. Building disappearing computers. *Communications of the ACM 48*, 3, 42–48.

SAPONAS, T. S., TAN, D. S., MORRIS, D., BALAKRISHNAN, R., TURNER, J., AND LANDAY, J. A. 2009. Enabling always-available input with muscle-computer interfaces. *UIST 2009*, 167–176.

SHARP, T., KESKIN, C., ROBERTSON, D., TAYLOR, J., SHOTTON, J., KIM, D., RHEMANN, C., LEICHTER, I., VINNIKOV, A., WEI, Y., FREEDMAN, D., KOHLI, P., KRUPKA, E., FITZGIBBON, A., AND IZADI, S. 2015. Accurate, robust, and flexible real-time hand tracking. *CHI 2015*, 3633—-3642.

SHOTTON, J., SHARP, T., KIPMAN, A., FITZGIBBON, A., FINOCCHIO, M., BLAKE, A., COOK, M., AND MOORE, R. 2013. Real-time human pose recognition in parts from single depth images. *Communications of the ACM 56*, 1, 116–124.

SHUEY, D., BAILEY, D., AND MORRISSEY, T. P. 1986. PHIGS: A standard, dynamic, interactive graphics interface. *IEEE Computer Graphics and Applications 6*, 8, 50–57.

SKOLNIK, M. I. 1962. Introduction to radar. *Radar Handbook 2*.

SMITH, C., AND GOGGANS, P. 1993. Radar target identification. *IEEE Antennas and Propagation Magazine 35*, 2 (April), 27–38.

SMITH, J., WHITE, T., DODGE, C., PARADISO, J., GERSHENFELD, N., AND ALLPORT, D. 1998. Electric field sensing for graphical interfaces. *IEEE Computer Graphics and Applications 18*, June, 54–59.

SMULDERS, P. 2002. Exploiting the 60 GHz band for local wireless multimedia access: Prospects and future directions. *IEEE Communications Magazine 40*, 1, 140–147.

SONG, J., SÖRÖS, G., PECE, F., FANELLO, S. R., IZADI, S., KESKIN, C., AND HILLIGES, O. 2014. In-air gestures around unmodified mobile devices. *UIST 2014*, 319–329.

STORCHEUS, D., ROSTAMIZADEH, A., AND KUMAR, S. 2015. A survey of modern questions and challenges in feature extraction. In *Proceedings of the 1st International Workshop on Feature Extraction: Modern Questions and Challenges, NIPS*, 1–18.

STOVE, A. G. 1992. Linear FMCW radar techniques. In *IEEE Proceedings on Radar and Signal Processing*, vol. 139, IET, 343–350.

STRICKON, J., AND PARADISO, J. 1998. Tracking hands above large interactive surfaces with a low-cost scanning laser rangefinder. *CHI'98*, 231–232.

SUTHERLAND, I. E., BLACKWELL, A., AND RODDEN, K. 1963. Sketchpad: A man-machine graphical communication system. Tech. Rep. TK296, Lincoln Laboratory, MIT, Lexington, MA.

WAN, Q., LI, Y., LI, C., AND PAL, R. 2014. Gesture recognition for smart home applications using portable radar sensors. In *36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2014)*, IEEE, 6414–6417.

WANG, Y., AND FATHY, A. E. 2011. Micro-doppler signatures for intelligent human gait recognition using a UWB impulse radar. In *IEEE Antennas and Propagation Society, AP-S International Symposium (Digest)*, 2103–2106.

WANG, R. Y., AND POPOVIĆ, J. 2009. Real-time hand-tracking with a color glove. In *SIGGRAPH 2009*, ACM Press, New York, New York, USA, 1.

WATSON-WATT, R. 1945. Radar in war and in peace. *Nature 155*, 3935, 319–324.

WEI, T., AND ZHANG, X. 2015. mTrack: High-precision passive tracking using millimeter wave radios. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, ACM, 117–129.

WEICHERT, F., BACHMANN, D., RUDAK, B., AND FISSELER, D. 2013. Analysis of the accuracy and robustness of the Leap Motion controller. *Sensors 13*, 5, 6380–6393.

WEIGEL, M., LU, T., BAILLY, G., OULASVIRTA, A., MAJIDI, C., AND STEIMLE, J. 2015. iSkin: Flexible, stretchable and visually customizable on-body touch sensors for mobile computing. In *CHI 2015*, ACM, 1–10.

WU, J., KONRAD, J., AND ISHWAR, P. 2013. Dynamic time warping for gesture-based user identification and authentication with Kinect. In *ICASSP 2013*, 2371–2375.

YATANI, K., AND TRUONG, K. N. 2012. BodyScope: A wearable acoustic sensor for activity recognition. In *Ubicomp '12*, ACM, 341–350.

ZHAI, S., MILGRAM, P., AND BUXTON, W. 1996. The influence of muscle groups on performance of multiple degree-of-freedom input. *CHI '96*, 308–315.

ZHUANG, Y., SONG, C., WANG, A., LIN, F., LI, Y., GU, C., LI, C., AND XU, W. 2015. SleepSense: Non-invasive sleep event recognition using an electromagnetic probe. In *IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN 2015)*, 1–6.