

# LIGHT CURTAIN LOCALIZATION VIA COMPRESSIVE SENSING

G. Pope<sup>1</sup>, M. Lerjen<sup>1</sup>, S. Müllener<sup>1</sup>, S. Schläpfer<sup>2</sup>, T. Walti<sup>1</sup>, J. Widmer<sup>2</sup>, and C. Studer<sup>3</sup>

<sup>1</sup>Dept. Inf. Tech. & EE, ETH Zurich, Switzerland

{gpope, mlerjen}@nari.ee.ethz.ch, {stevenm, twalti}@student.ethz.ch

<sup>2</sup>{simon.schlaepfer, widmerjo}@gmail.com

<sup>3</sup>Dept. ECE, Rice University, Houston, TX, USA, studer@rice.edu

## ABSTRACT

Light curtains are presence detection devices, typically employed to detect when an object enters (or passes through) a region to initiate emergency security procedures. However, most existing light curtain implementations are not designed to detect the precise location where the barrier was broken. In this paper, we present a hardware prototype implementation that is able to identify the locations where the light curtain was broken, even if multiple objects penetrate the barrier simultaneously. To this end, we deploy techniques from sparse signal recovery and compressive sensing to perform detection and localization with only a few infrared transmitters and sensors. The proposed prototype implementation is scalable to large physical sizes and can be tailored to suit a particular application in terms of the number and size of objects to detect, as well as the desired spatial resolution.

**Index Terms**— Compressive sensing, hardware prototype, localization, light curtains, sparse signal recovery.

## 1. INTRODUCTION

Significant theoretical and algorithmic progress has been made in recent years in the fields of sparse signal recovery and compressive sensing (CS) [1, 2]. However, apart from expensive magnetic resonance imaging (MRI) devices, there have been only a few physical implementations of systems described in the literature that make use of CS. Notable designs are the single-pixel camera [3], Xampling to recover wideband signals [4], a portable ECG monitor [5], an ASIC for real-time audio signal restoration [6], and a capacitive touch screen [7]. Light curtains (see, e.g., [8] for more details) offer a new application field which could benefit from CS. Such devices use optical transmitters and sensors to detect whether an object enters (or passes through) a predefined region to initiate certain actions. For instance, light curtains can be used to detect if a person enters an area with heavy machinery to initiate an emergency shutdown procedure.

---

This work was supported in part by the Swiss National Science Foundation under Grant PA00P2-134155. The authors would like to thank H. Bölcskei for his support and A. Burg for inspiring discussions.

## 1.1. Contributions

This paper develops a novel localization method for light curtain implementations relying on CS. The method is able to localize objects that enter a region of interest (ROI) using only a few infrared (IR) transmitters and receivers. The key idea of our approach is “sparsity,” i.e., the number of objects placed within the ROI (or, more precisely, the area occupied by all objects) is small compared to the overall area. Unfortunately, optimal localization turns out to be a non-linear and non-convex optimization problem, but we show that linearization enables us to reformulate it as a standard CS recovery problem. It turns out the localization complexity is proportional only to the number of receivers (and not the number of transmitters). We finally show a hardware prototype implementation of the proposed CS-based light curtain. With this prototype, we demonstrate the detection and localization capabilities for applications where a small number of objects breaks the light barrier.

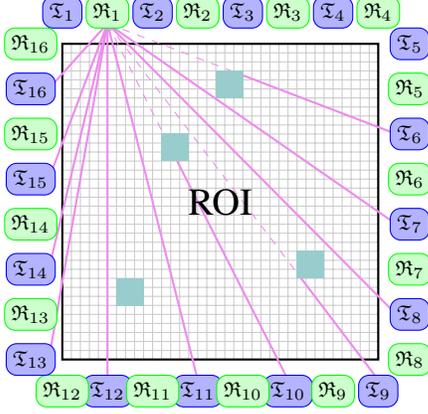
## 1.2. Prior work

Existing light curtains, such as [8] are designed primarily to detect only if the ROI is penetrated and, as such, are incapable of detecting where the intrusion occurred. To apply existing multi-touch technologies to light curtain designs, e.g., to detect multiple objects, one needs a “surface” that an object can pass through. As such, resistive or capacitive technologies, as well as surface acoustic wave based methods [7, 9] are not applicable. The product [10] could be employed for our application, although no information about the underlying technology is available. Thus, we propose an open design for localizable light curtains that use CS-based techniques to locate a number of objects with low hardware overhead.

## 2. SYSTEM DESCRIPTION

### 2.1. System model

We assume that the ROI (where we want to detect and localize objects) is surrounded by  $n_r$  receivers  $\{\mathfrak{R}_1, \dots, \mathfrak{R}_{n_r}\}$  and  $n_t$



**Fig. 1.** Illustration of the transmitters ( $\mathcal{T}_i$ ), receivers ( $\mathcal{R}_j$ ), and four objects in the ROI (shown in teal). The paths from each of the transmitters to  $\mathcal{R}_1$  are shown in pink; note how the objects block some of these paths.

transmitters  $\{\mathcal{T}_1, \dots, \mathcal{T}_{n_t}\}$ , as shown in Fig. 1. Assume that the  $i^{\text{th}}$  transmitter  $\mathcal{T}_i$  transmits a signal with power  $p_i$ , so if no obstructions and reflections are present, the  $j^{\text{th}}$  receiver  $\mathcal{R}_j$  will receive a signal with power  $r_j = \sum_{i=1}^{n_t} \omega_{i,j} p_i$ , where  $\omega_{i,j}$  is a weighting factor (possibly equal to zero) to compensate for signal attenuation due to distance and angle sensitivity of the transmitter and the receiver. If an object is placed somewhere on the line between  $\mathcal{T}_i$  and  $\mathcal{R}_j$ , this receiver will not measure any power from  $\mathcal{T}_i$ . Let  $\mathcal{X}_{i \rightarrow j}$  be the set of grid cells, so that if an object was placed on any of them, it would block the ray between  $\mathcal{T}_i$  and  $\mathcal{R}_j$ . Let  $\{p_1, \dots, p_{n_t}\}$  be the set of transmit power values, then the power received at  $\mathcal{R}_j$  is given by

$$r_j = \sum_{i=1}^{n_t} \omega_{i,j} \tilde{p}_{i,j}, \quad (1)$$

where  $\tilde{p}_{i,j} = p_i$  if no object lies on  $\mathcal{X}_{i \rightarrow j}$  and  $\tilde{p}_{i,j} = 0$  otherwise.

The goal is now to find the locations of the objects given the set of observed power levels  $\{r_1, \dots, r_{n_r}\}$  and the locations of the transmitters and receivers, as well as the transmit-power levels. Doing so is, however, non-trivial as one must solve the following optimization problem:

$$\begin{aligned} & \text{minimize} && \text{the number of objects present} \\ & \text{subject to} && \text{them generating the observations as in (1).} \end{aligned}$$

Unfortunately, this is a combinatorial problem as we search for the smallest number of objects explaining the observation.

## 2.2. Linearization

To arrive at an efficient way of identifying the locations of the objects in the ROI, we propose to linearize (1). To do so, let  $\mathbf{x} \in \mathbb{R}^n$  be a vector which models if an object is present at a grid point. If an object is at location  $k$  we assume that

$x_k > 0$ , but if no object is present, then  $x_k = 0$ . With this, we can reformulate (1) as

$$r_j = \sum_{i=1}^{n_t} \omega_{i,j} p_i \left( 1 - \sum_{k \in \mathcal{X}_{i \rightarrow j}} x_k \right) = \sum_{i=1}^{n_t} \omega_{i,j} p_i - \sum_{i=1}^{n_t} \sum_{k \in \mathcal{X}_{i \rightarrow j}} \omega_{i,j} p_i x_k,$$

which can be written as  $\mathbf{r} = \mathbf{D}\mathbf{x} + \mathbf{p}$ , where  $\mathbf{r} \in \mathbb{R}^{n_r}$  is the vector of received power levels,  $\mathbf{D} \in \mathbb{R}^{n_r \times n}$  is the effective measurement matrix,  $\mathbf{x} \in \mathbb{R}^n$  is the vector of  $x_k$ 's and  $\mathbf{p} \in \mathbb{R}^{n_r}$  is the vector containing the signals that would be received if no object was present. The  $(j, k)$ th entry of  $\mathbf{D}$  is given by  $-\omega_{i,j} p_i$  if  $k \in \mathcal{X}_{i \rightarrow j}$ , that is, if  $x_k$  lies between the  $i^{\text{th}}$  transmitter and the  $j^{\text{th}}$  receiver, and 0 otherwise. The  $j^{\text{th}}$  component of  $\mathbf{p}$  is given by  $p_j = \sum_{i=1}^{n_t} \omega_{i,j} p_i$ . As there will be some noise in the system (e.g., caused by the IR receivers and analog circuitry, as well as interference from other light sources), we include an additive noise term  $\mathbf{n} \in \mathbb{R}^{n_r}$ , giving the final (linearized) system model

$$\mathbf{r} = \mathbf{D}\mathbf{x} + \mathbf{p} + \mathbf{n}. \quad (2)$$

In order to identify the locations of the objects within the ROI, one could solve the following P0 problem

$$\underset{\hat{\mathbf{x}}}{\text{minimize}} \quad \|\hat{\mathbf{x}}\|_0 \quad \text{subject to} \quad \|\mathbf{r} - \mathbf{p} - \mathbf{D}\hat{\mathbf{x}}\|_2 \leq \varepsilon$$

where  $\|\mathbf{a}\|_0$  is the number of non-zero entries in  $\mathbf{a}$  and for some  $\varepsilon \geq \|\mathbf{n}\|_2$ . This problem, however, requires a combinatorial search and hence, we resort to its convex relaxation known as basis pursuit denoising (BPDN) [2, 11, 12]

$$\underset{\hat{\mathbf{x}}}{\text{minimize}} \quad \|\hat{\mathbf{x}}\|_1 \quad \text{subject to} \quad \|\mathbf{r} - \mathbf{p} - \mathbf{D}\hat{\mathbf{x}}\|_2 \leq \varepsilon,$$

to arrive at an efficient way of localizing objects in the ROI.

## 2.3. Low-complexity localization algorithm

There are many alternatives to BPDN that could be used, such as orthogonal matching pursuit (OMP) [13–15] or CoSaMP [16], both of which have the advantage that no estimate of the noise power  $\varepsilon$  is required. Although such algorithms can be implemented efficiently in hardware (e.g., [6, 17]), we seek even faster and simpler algorithms, as we are primarily interested in a low-complexity real-time solution. Since we are not interested in the actual signal values (only whether the entry is zero or not), we propose to use a one-pass thresholding procedure as in [18, Alg. 1], which finds the columns of the dictionary that are most correlated with the observation. Concretely, we calculate the signal estimate

$$\hat{\mathbf{x}} = \mathbf{D}^T \mathbf{y} = \mathbf{D}^T \mathbf{D} \mathbf{x} + \mathbf{D}^T \mathbf{n}, \quad (3)$$

with  $\mathbf{y} = \mathbf{r} - \mathbf{p}$  and then find the locations of local maxima, which should correspond to the location of an object because  $\mathbf{D}^T \mathbf{D}$  is diagonal dominant for well-designed light curtain devices (see Sec. 2.4).

Figure 3 shows an example output of this correlation procedure. Here we have mapped the vector  $\mathbf{D}^T \mathbf{y}$  onto a  $50 \times 50$  grid (so that  $n = 2500$ ). In this case, we have simulated three objects being placed on the grid, which correspond to the three peaks. By using this correlation method, we can reliably estimate the locations of the objects with a single matrix-vector multiplication, whereas BPDN, OMP, or CoSaMP require considerably more operations to form a signal estimate.

#### 2.4. Measurements & more measurements

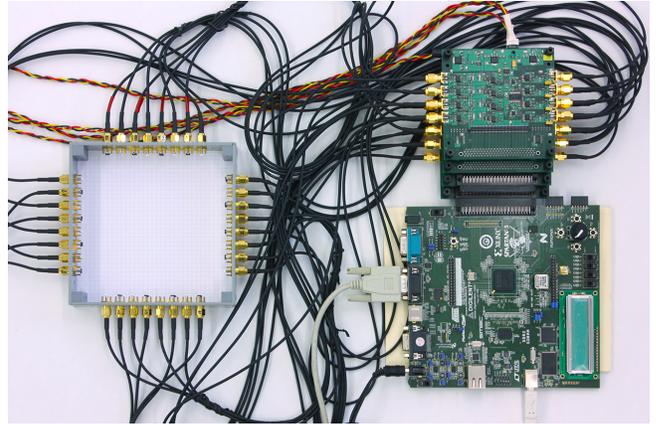
Since the actual measurement matrix  $\mathbf{D}$  is a non-trivial function of the transmit power levels and of the locations of the transmitters and receivers, there is no straightforward way of optimizing the power levels to generate a  $\mathbf{D}$  with “good” properties. Thus, inspired by sensing matrices typically used for CS, e.g., [1, 2, 13], we simply choose the transmit powers from a random distribution, such as a Gaussian or Bernoulli distribution. To transmit signals with “negative power”, we perform two sets of measurements resulting in two matrices  $\mathbf{D}^+$  and  $\mathbf{D}^-$ ; the first corresponding to the positive transmit signals and the second to the negative transmit signals. Thus, by performing two measurement passes, one using only the positive transmit signals and the other using only the negative transmit signals, and then take the difference of the two observations, we arrive at the effective dictionary  $\mathbf{D} = \mathbf{D}^+ - \mathbf{D}^-$ .

The idea of taking two consecutive measurements can be carried even further. If the time taken to perform one measurement at each receiver is small compared to the speed of the objects, then we should be able to perform multiple measurements, while the objects still occupy the same positions. Furthermore, we should be able to perform each of these measurements with *different* transmit power levels and then combine these measurements to improve the localization performance. Combining  $T$  consecutive measurements can be done by stacking the observations as follows:

$$\mathbf{r} = \begin{bmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_T \end{bmatrix} = \begin{bmatrix} \mathbf{D}_1 \\ \vdots \\ \mathbf{D}_T \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_T \end{bmatrix} + \mathbf{n}, \quad (4)$$

and then solving this new system for  $\mathbf{x}$ .

It is worth briefly discussing what happens if  $T$  is equal to the number of transmitters. In this case, it would be possible to have only one transmitter active for each measurement, instead of using random power levels at all transmitters. In this case, we can still find the locations of the objects, with slightly improved performance, see Fig. 4. This approach, however, requires a search through all possible grid locations to find the smallest number of objects required to generate the given observations, which is considerably more complex than our correlation-based approach. Moreover, a large number of measurements are required if the number of transmitters  $T$  is large. We finally note that even if  $T$  is equal to the number



**Fig. 2.** Prototype: on the left is the frame for the ROI ( $145 \text{ mm} \times 145 \text{ mm}$ ) in which we detect any objects. We place four transmitters and four receivers on each side, the PCBs on the top right contain the DACs and ADCs and the PCB on the bottom right hosts the FPGA control circuitry.

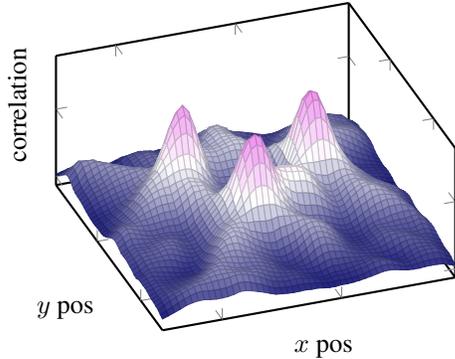
of transmitters, (4) is still an undetermined system as  $\mathbf{D}$  is an  $Tn_r \times n$  matrix, where  $n \gg Tn_r$  is the number of grid points.

The final issue is to decide how many transmitter and receivers one must choose and where to place them. Again, optimizing the placement so as to produce a “good” matrix  $\mathbf{D}$  is a hard problem. However, it is intuitively apparent that it is necessary to have a different set of beams covering each grid cell. We found by extensive simulations that an equal number of transmitters and receivers on all four edges of the grid, with a slight clustering towards the corners, performs well. Additional simulations furthermore show that the number of transmitters and receivers required is roughly proportional to the perimeter of the region we are interested in.

Finally, since each receiver corresponds to a row of  $\mathbf{D}$ , adding more receivers increases the computational complexity of finding  $\mathbf{x}$ . The number of transmitters, however, does not directly correspond to any dimension parameter of  $\mathbf{D}$ . Therefore, one can easily increase the number of transmitters, with the intention of getting better coverage of the grid and hence, better recovery performance, with no extra computational cost (beyond the one-off cost of precomputing the entries of the matrix  $\mathbf{D}$ ).

### 3. HARDWARE IMPLEMENTATION

In order to demonstrate the efficacy of our localization approach, we developed a light curtain prototype implementation shown in Fig. 2. For this prototype, the ROI is  $145 \text{ mm} \times 145 \text{ mm}$ , contained within a frame (shown on the left). The prototype uses 16 transmitters and 16 receivers, placed alternately on each side of the ROI. As transmitters we use IR LEDs and photo-diodes as receivers, connected to 16 bit DACs and 12 bit ADCs, running at 640 kHz and 800 kHz, respectively. We use readily available, off-the-shelf



**Fig. 3.** Correlation map calculated for three objects placed on a  $50 \times 50$  grid using 16 transmitters and 16 receivers. The position of the three objects is readily apparent and given by the three dominant local maxima.

components. These components are connected to custom built PCBs (top right of Fig. 2), which are then connected to the FPGA controller board (Xilinx Spartan 3-AN Starter Kit, shown on the bottom right). Recovery is currently performed by a host PC running Matlab. Implementing the correlation-based localization and detection algorithm directly on the FPGA is part of on-going work.

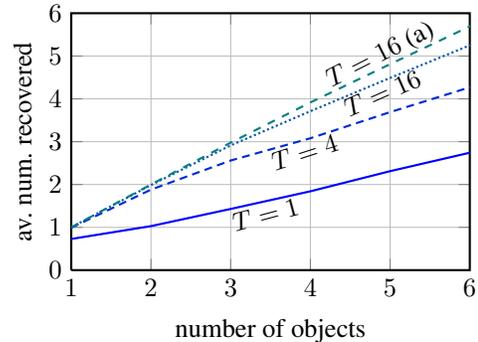
To avoid interference from ambient light sources, we send a 20 kHz sine wave with maximum amplitude  $p_i$  for 0.64 ms, instead of transmitting a constant signal. At the receiver we then filter the signal with a bandpass filter removing interference caused by sources such as natural light and fluorescent lamps. The signals are sampled at 800 kHz and the root mean square value is calculated across 512 samples, before being sent to the host PC. Ignoring data transfer, the prototype can take a maximum of 1 500 measurements per second.

## 4. RESULTS

### 4.1. Simulation results

We begin by modeling the system in software. To do this, we assume the sensor placement shown in Fig. 1 which consists of 16 transmitters and 16 receivers uniformly placed around the four sides. We assume that we want to resolve the objects on a  $50 \times 50$  grid within the sensor array. We then randomly place objects occupying a  $3 \times 3$  block on the grid. For this set of object positions, we calculate the received signals  $r_1, \dots, r_T$  corresponding to  $T$  measurements of the system with random transmit values and then calculate the dictionary  $\mathbf{D}$  off-line. Finally, reconstruction is carried out as in (3) followed by detecting local maxima, which should be the positions of the objects. This procedure is repeated 1 000 times, each time with different object positions and transmit powers. An example of the resulting correlations for three objects is shown in Fig. 3, where we see that the three objects are clearly distinguished using the correlation map.

In Fig. 4 we show the average number of objects recover-



**Fig. 4.** Average number of objects recoverable for differing numbers of measurements  $T$ . For  $T = 1, 4, 16$  the transmit powers are drawn from a uniform distribution and for  $T = 16(a)$ , only one transmitter is active for each measurement.

able, for several values of  $T$ . By “recover” we mean that we can find the center of the object within a radius of 3 grid points from its original position. For the case when  $T$  equals the number of transmitters, we compare two transmit schemes, one where the transmit values are chosen uniformly, the other where only one transmitter is active for each measurement, the latter of which, performs marginally better where we can almost always recover 4 of 4 objects.

### 4.2. Prototype performance

Our light curtain prototype is able to accurately resolve a small number of objects that are placed within the ROI. We divided the  $145 \text{ mm} \times 145 \text{ mm}$  ROI into a  $20 \times 20$  grid and used  $T = 16$  measurements. We were able to consistently resolve up to 3 objects placed in the ROI, after which the performance dropped off.<sup>1</sup> We found the best results were achieved with only a small number of transmitters active, each transmitting at full power (corresponding to a Bernoulli distribution).

## 5. CONCLUSION

We have demonstrated the use of CS techniques to create an IR light curtain that is able to localize several objects with low hardware overhead and low computational complexity. Our simulations and prototype implementation have demonstrated the potential of our approach. The proposed architecture can be scaled to large physical dimensions and three dimensions, and so could be effectively employed to guard large spaces. On a much smaller scale, our prototype could be used as a multi-touch trackpad that can be mounted to an existing display similar to that in [10].

There are many avenues for further work, such as implementing our localization algorithm on the FPGA and mounting a large display behind the frame to facilitate testing.

<sup>1</sup>We are in the process of mounting a tablet PC behind the frame so as to compare the object locations and numerically quantify the error.

## 6. REFERENCES

- [1] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, Mar. 2008.
- [2] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [3] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Processing Magazine*, pp. 83–91, Mar. 2008.
- [4] M. Mishali, Y. C. Eldar, O. Dounaevsky, and E. Shoshan, "Xampling: Analog to digital at subnyquist rates," *Circuits, Devices & Systems, IET*, vol. 5, no. 1, pp. 8–20, Jan. 2011.
- [5] K. Kanoun, H. Mamaghanian, N. Khaled, and D. Atienza, "A real-time compressed sensing-based personal electrocardiogram monitoring system," in *Proc. of the IEEE/ACM 2011 Design, Automation and Test in Europe Conference*, Grenoble, France, Mar. 2011.
- [6] P. Maechler, C. Studer, D. Bellasi, A. Maleki, A. Burg, N. Felber, H. Kaeslin, and R. G. Baraniuk, "VLSI design of approximate message passing for signal restoration and compressive sensing," *IEEE J. on Emerg. Sel. Topics in Circ. and Sys.*, vol. 2, no. 3, pp. 579–590, Oct. 2012.
- [7] C. Luo, M. A. Borkar, A. J. Redfern, and J. H. McClellan, "Compressive sensing for sparse touch detection on capacitive touch screens," *IEEE J. on Emerg. Sel. Topics in Circ. and Sys.*, vol. 2, no. 3, pp. 639–648, Sep. 2012.
- [8] Banner, "Ez-screen safety light curtain systems," <http://www.bannerengineering.com/en-US/products/6/Machine-Safety/45/Safety-Light-Curtains-Type-4/25/>, Nov. 2012.
- [9] M. R. Bhalla and A. V. Bhalla, "Comparative study of various touchscreen technologies," *Int. J. Comp. Appl.*, vol. 6, no. 8, pp. 12–18, Sep. 2010.
- [10] P. Labs, "G4 touch screen," <http://multi-touch-screen.com/product.html>, Nov. 2012. [Online]. Available: <http://multi-touch-screen.com/product.html>
- [11] J.-J. Fuchs, "Recovery of exact sparse representations in the presence of bounded noise," *IEEE Trans. Inf. Theory*, vol. 51, no. 10, pp. 3601–3608, Oct. 2005.
- [12] T. Cai, L. Wany, and G. Xu, "Stable recovery of sparse signals and an oracle inequality," *IEEE Trans. Inf. Theory*, vol. 56, no. 7, pp. 3516–3522, July 2010.
- [13] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.
- [14] Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. of 27th Asilomar Conf. on Signals, Systems, and Comput.*, Pacific Grove, CA, USA, Nov 1993, pp. 40–44.
- [15] S. G. Mallat, G. Davis, and Z. Zhang, "Adaptive time-frequency decompositions," *SPIE Journal of Optical Engineering*, vol. 33, pp. 2183–2191, July 1994.
- [16] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *App. Comp. Harm. Anal.*, vol. 26, pp. 301–321, 2008.
- [17] P. Maechler, P. Greisen, B. Sporrer, S. Steiner, N. Felber, and A. Burg, "Implementation of greedy algorithms for LTE sparse channel estimation," in *Proc. 44th Asilomar Conf. Signals, Systems and Computers.*, Pacific Grove, CA, USA, Nov. 2010, pp. 400–405.
- [18] W. U. Bajwa, R. Calderbank, and S. Jafarpour, "Why Gabor frames? Two fundamental measures of coherence and their role in model selection," *J. Commun. Netw.*, vol. 12, no. 4, pp. 289–307, Aug. 2010.