Super-resolution Sparse Projected Capacitive Multitouch Sensing

Mehrdad Yaghoobi†**, Stephen McLaughlin***, **Mike E.Davies**†

† Institute for Digital Communications, The University of Edinburgh, EH9 3JL, UK
* School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh EH14 4AS, UK

Keywords: Sparse Signal Models, Multitouch, Touchscreens, Super-resolution, Structured Sparse Signals

Abstract

The resolution of capacitive touch sensors is not sufficient for some sensing applications. The coarse resolution of a sensor with a regular grid is limited by the half distance between two adjacent layer-crossovers. Increasing the density of such layer-crossovers, improves the resolution of touch signals. While this technique needs some modification in the hardware, it does not necessarily guarantee that the corresponding estimated touch locations are more accurate.

We explore the problem of resolution enhancement for the multitouch sensors here, using the sparsity pattern of the touch signals. This is called the *super-resolution* as the goal is to enhance the sensor resolution, while not changing the sensing hardware, and only incorporates some prior information about the input. The super-resolution problem can be computationally very costly. The aim is to present a rather simple algorithm to run in the real-time. This has been achieved by exploiting a structured sparse representation, which allows a computationally simple super-resolution algorithm.

1 Introduction

Touch sensors are becoming more accessible in modern electronic devices. The touch sensors are now embedded into the screens to easily capture the human finger touches, as an input to the systems. There has been a significant progress in designing various hardwares to capture such finger touches [16], where *capacitive touch sensors* are among the most successful and widely used sensors. Some of these sensors can handle multitouch sensing by detecting the finger touches on a regular grid of sensors. As the locations of the finger touches are arbitrary located on the surface of the sensor, the touch locations are actually in the continuous domain. The process of touch sensing with a regular grid, can be interpreted as the sampling of such touch signals. The process of reconstruction of the original touch signal using the Whittaker-Shannon interpolation formula is not very efficient, as the input signal does not satisfy the necessary condition of exact reconstruction in the classical Shannon-Nyquist sampling theorem.

The *resolution of the touch sensor* is defined as the smallest identifiable change in the touch locations. For the single touch sensing, there exist techniques to improve the resolution by interpolating the values of the adjacent sensing locations [14]. These techniques essentially rely on the fact that there exists only a **single touch**. In a general setting of multitouch sensing, these techniques are not successful. On the other hand, the problem of resolution enhancement in this setting is indeed very ill-posed, if we do not incorporate any extra information about the input.

The touch signals generally have some sparse structures. The reason is that we are normally looking for a certain number of touches, which can vary in different devices. Such signals actually have limited degrees of freedom, or a *finite rate of innovation* (FRI) [18]. They can be recovered using the Prony's method, and solving a Vandermonde system, which is a bad-conditioned linear system. Recovering the signal using FRI model is thus sensitive to the input noise. A noise resilient version of the FRI recovery algorithm is presented in [8], which uses an iterative denoising algorithm, called the Cadzow's iterative denoising [3]. This technique is based on iterative projections on the rank-K and Toeplitz matrix sets, which needs some Singular Value Decompositions and it is thus computationally very expensive to our application.

1.1 Prior Work

The problem of resolution enhancement for capacitive touch sensors for single touches, was investigated in [14]. This technique will briefly be explored in Section 3.1. Although an extension of this interpolation method, using a more complicated model for the touch sensors, is possible, see for example [10], here we use a sparse model for touch signals to not only improve the resolution of *single touch* signals, but also **to consider a more general multitouch scenario**.

The problem of recovering signals with a FRI, which has been explored in [18], can be used with Gaussian sampling kernel [8]. Such a sampling strategy well fits to the touch sensing application. By adding the computational cost of denoising, the computational cost of algorithm grows up significantly. We therefore chose a different approach, which has similarities with the single-exposure super-resolution, using a sparsity model for the signals [4]. Specifically, we use a similar structured sparsity model to solve a sensing problem with a coherent linear sensing operator. Moreover, we present two practical algorithms for touch location identification, which do not involve any infinite dimensional optimisation problems, while considering the positivity of touches.

The idea of using sparsity in the capacitive touch sensors, has also recently been used for the *compressive touch sensing* in [11, 12]. Luo *et al.* proposed a technique to accelerate the sensing process, which needs some sensor hardware modifications, *i.e.* electrically charging a row of sensors by some random values and measuring the total charge. Here we propose an essentially different technique to improve the resolution of current sensors, *without changing the sensor hardware*.

1.2 Contribution

In this paper, we investigate the problem of single-exposure *super-resolution multitouch sensing*. We initially formulate the touch signals by a linear additive model, with the Gaussian type elementary functions. This model is inspired from real data observation. Such a model can be used for canonical sparse approximation, which would not be very successful, as the generative model is highly coherent, *i.e.* many elementary functions are similar and distinguishing each of them from the neighbours, is difficult. We incorporate a *separate touch signal model* to assist the sparse recovery. The new sparse signal model has been used to obtain super-resolution multitouch sensing. Although the standard sparse approximation has already been used for the image super-resolution [20], it will fail to recover an (approximate) super-resolution signal using a coherent signal model. We here show that the canonical barriers in such an image super-resolution can be moved using a **structured sparsity model**.

2 Problem Formulation

We assume that the grid is rectangular and regular, *i.e.* equidistant, and each finger touch triggers more than a single layer-crossover. This is what normally happens, as the size of fingers are larger than the distance between two layer-crossovers. In Figure 1, we show the signal received using a typical 3.5' capacitive touchpad of size 9×12, which shows a set of points on the grid, triggered by a single finger touch. We assume that each touch can be modelled as a 2D Gaussian function, which is roughly what is observed in Figure 1.

The capacitive change caused by a set of touches, can be modelled in the continuous domain by,

$$t(x, y) \approx \sum_{1 \leq i \leq K} a_i e^{\frac{(x-x_i)^2+(y-y_i)^2}{\sigma^2}} \qquad (1)$$

where K is the number of touches and (x_i, y_i) is the centre of the i^{th} touch. If the grid is $M \times N$ and, the distance between two adjacent layer-crossovers, which is also called the *pitch spacing*, is Δ, the centre of touches are located in $((0.5+m)\Delta, (0.5+n)\Delta)$, where $m \in [0, M]$ and $n \in [0, N]$. We sense, *i.e.* sample, t on this grid, which gives us MN samples $\{t(m, n)\}_{1 \leq m \leq M, 1 \leq n \leq N}$. Sampling of $t(x, y)$, with the capacitive touch sensors, can be seen as the convolution of $d(x, y) = \sum_{1 \leq i \leq K} a_i \delta_{x_i, x_j}(x, y)$, where $\delta_{x_i, y_i}(x, y)$ is the delta Dirac at the location (x_i, y_i), with the Gaussian kernel $g(x, y) \triangleq e^{\frac{x^2+y^2}{\sigma^2}}$. We can now

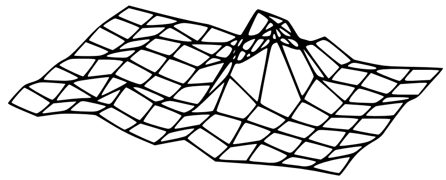

Figure 1: The sensed signal by a 9×12 capacitive touch sensor, when a single finger touches the sensor.

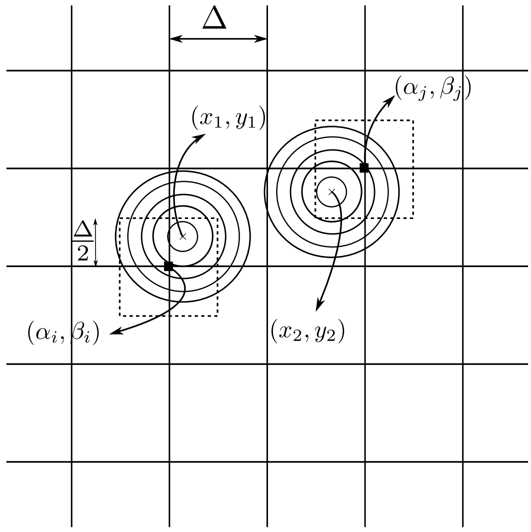

Figure 2: A schematic diagram of a part of touch-sensing grid. Two touches happen at locations (x_1, y_1) and (x_2, y_2).

represent the sampling operator $\mathcal{T}_g$, which is the sampling of the convolution operation with the kernel g, as follows,

$$\mathbf{z} = \mathcal{T}_g d(x, y), \ \mathbf{z} \in \mathbb{R}^{MN}. \qquad (2)$$

The task is now to recover $d(x, y)$, or only (x_i, y_i)'s, as accurate as possible, given the sensed vector $\mathbf{z}$. As the delta Dirac functions $\delta_{x_i, y_i}(x, y)$'s can arbitrary locate at any place in $\left((\frac{\Delta}{2}, \Delta M + \frac{\Delta}{2}), (\frac{\Delta}{2}, \Delta N + \frac{\Delta}{2})\right)$, the resolution of the recovered locations (x_i, y_i) is bounded by $\frac{\Delta}{2}$, using a Whittaker-Shannon type interpolation formula.

The process of finding the locations of touches, is often simplified by *finding the closest points on the grid*, assuming that each layer-crossover has only one finger touch in its neighbourhood. Such an assumption is necessary for a successful multitouch sensing, as violating this condition, can cause two different touches be found at a single layer-crossover. A schematic diagram of a part of a *regular grid* touch-sensor, when two touches occur at the locations (x_1, y_1) and (x_2, y_2), is drawn in Figure 2. The neighbourhood of the layer-crossovers (α_i, β_i) and (α_j, β_j) are shown by the dashed boxes in this figure.

2.1 Peak Picking for Touch Sensing

An intuitive method to identify the locations of touches is to find the peaks on the sensor grid [14]. Here, we assume that touches are far from each other, which improves the success of identification of the delta Dirac locations.

2.2 Sparse Signal Model for Touch Sensing

An alternative approach to boost the performance of peak picking, is to incorporate the fact that each finger-touch can actually be modelled with a Gaussian shape signal. We can also incorporate the fact that the magnitudes a_i's are all positive.

When we assume a Gaussian model for each touch, a good *approximate signal model* is a discrete Gaussian, centred on top of each layer-crossover. Let $\mathbf{D}$ be a matrix with the $(i,j)^{th}$ element equal to $\gamma g(\alpha_i - \alpha_j, \beta_i - \beta_j)$, which represents the discrete 2D Gaussian functions centred on the grid and $\gamma = \sqrt{\frac{2}{\pi\sigma^2}}$ is the normalisation factor. We here try to find a θ such that observed signal $\mathbf{z}$ is close to $\mathbf{D}\theta$. For the multitouch sensing problem, $d(x,y)$ is sparse, *i.e.* $K \ll MN$. We therefore seek a K-sparse θ. It is worth mentioning that $\mathbf{D} \in \mathbb{R}^{MN \times MN}$, which is called a *dictionary*, is a square matrix, the sparsity assumption of θ is necessary as $\mathbf{D}$ is mostly ill-conditioned and the observation is often noisy. The touch locations can thus be found by solving the following program,

$$\min_{\theta \in \mathcal{C}} \|\mathbf{z} - \mathbf{D}\theta\|_2^2, \ \mathrm{s.t.} \ \|\theta\|_0 \leq K \tag{3}$$

where $\|\cdot\|_0$ counts the non-zero elements of the operand and the admissible set $\mathcal{C}$ is $\mathbb{R}_+^{MN}$. The active elements of θ, represent the possible locations of the touches *on the grid*. Solving (3) is computationally expensive, which can approximately be solved some greedy methods, e.g. Matching Pursuit (MP) [13]. A simple alternative is hard-thresholding, which is keeping the K largest coefficients of $\mathbf{D}^T\mathbf{z}$, and letting the rest to be zero [7]. A key characteristic of the dictionary is its *coherence*, which is defined as $\mu = \max_{i,j} |\mathbf{d}_i^T \mathbf{d}_j|$ [17]. The coherence of a dictionary actually shows that how similar are the columns of the dictionary, which are called the *atoms*. Showing the exact touch locations recovery using the coherence based analysis [9, 17] is not possible here, as the dictionary has a very high coherence, which makes the exact recovery almost impossible. We here need to incorporate some extra signal models.

If two (x_i, y_i)'s are very close, the recovery is more difficult and noise sensitive, as the corresponding closest (α_i, β_i)'s are very similar, *i.e.* $|\mathbf{d}_i^T \mathbf{d}_j| \approx 1$ for those atoms. Fortunately, this case is not happening here, because of the anatomy of human fingers. If we name the minimum distance of two delta Dirac R, a reasonable assume is that R is greater than 2Δ, which is also the assumption in [4] for super-resolution spectral sensing, to derive the exact recovery of input sparse spectrum. Considering the fact that there is a distance R between two non-zero elements of θ, we can reduce the error of touch locating, using a new admissible set $\mathcal{C}$, which excludes such unwanted sparse vectors. This type of sparse approximations is called *Model Based* Sparse Approximations [1].

3 Super-resolution Touch Sensing

The resolution achieved by finding the closest points on the grid, which is called the *coarse resolution* of the sensor, is not satisfactory for many applications. Super-resolution techniques are introduced to compensate the errors caused by the grid discretization, or simply reduce the aliasing effect. Here, these techniques are based on allowing the centres of touches locate off the sensing gird. We investigate two different approaches to super-resolution, in the next two subsections.

3.1 Linear Interpolation for Resolution Enhancement

The coarse resolution of peak picking can be improved by an interpolation post processing, e.g. [10]. O'Conner suggests a simple *linear* interpolation technique for resolution enhancement in [14]. In his method, we first find the peaks and their four neighbour points on the grid. Let (α_i, β_i) be the i^{th} detected peak and $\mathbf{z}_{(\alpha_i,\beta_i)}$ be the corresponding sensed value. We can now interpolate between the sensed points and find a new centre for the touch, using the following formula[1]:

$$\begin{aligned} \hat{x}_i &= \alpha_i + \frac{\Delta}{2}\left(\frac{\mathbf{z}_{(\alpha_i+1,\beta_i)} - \mathbf{z}_{(\alpha_i-1,\beta_i)}}{\zeta_i^x}\right) \\ \hat{y}_i &= \beta_i + \frac{\Delta}{2}\left(\frac{\mathbf{z}_{(\alpha_i,\beta_i+1)} - \mathbf{z}_{(\alpha_i,\beta_i-1)}}{\zeta_i^y}\right) \end{aligned} \tag{4}$$

where $\zeta_i^x = \max(|\mathbf{z}_{(\alpha_i,\beta_i)} - \mathbf{z}_{(\alpha_i+1,\beta_i)}|, |\mathbf{z}_{(\alpha_i,\beta_i)} - \mathbf{z}_{(\alpha_i-1,\beta_i)}|)$ and $\zeta_i^y = \max(|\mathbf{z}_{(\alpha_i,\beta_i)} - \mathbf{z}_{(\alpha_i,\beta_i+1)}|, |\mathbf{z}_{(\alpha_i,\beta_i)} - \mathbf{z}_{(\alpha_i,\beta_i-1)}|)$. This interpolation is based on a piecewise linear model for the touches. More accurate interpolation are possible using conventional non-linear interpolations, *e.g.* Cubic splines [10].

3.2 Super-resolution Touch Sensing using the Sparsity Model

An approach to improve the resolution of sparsity based touch sensing is to assume that (x_i, y_i)'s are located on a finer grid than the sensing grid. This type of sparsity based super-resolution is called the single frame, or single exposure, super-resolution [5,6]. In this setting, we can reformulate (3) using an overcomplete dictionary $\mathbf{D} \in \mathbb{R}^{MN \times P}$, where $P > MN$. If we generate a $\mathbf{D}$ over a regular grid, the centres of the Gaussian functions locate on some $(\hat{\alpha}_i, \hat{\beta}_i)$'s, where $1 \leq i \leq P$. If the sparse vector $\theta \in \mathbb{R}_+^P$, we need to solve the problem (3) with the overcomplete dictionary $\mathbf{D}$ to identify the possible touch locations.

As there exist correlated atom pairs in $\mathbf{D}$, *i.e.* $\mathbf{D}$ has a large μ, solving (3) can essentially fail in recovering the desired locations, if the number of touches is more than one and we do not incorporate extra prior information. We again restrict the locations of non-zero elements in $\mathcal{C}$ to be not closer than R. We will empirically demonstrate that the new formulation indeed helps us to reduce the location recovery error.

[1] The formulation here is slightly different to [14].

4 Algorithms

Solving (3) is more difficult than a sparse approximation, as we also need to consider the non-negativity and the spacial constraint. The algorithm should also be computationally cheap, as the aim is to find a fast method for real-time implementation. The most computationally expensive operation in the spars approximation methods is the calculation of dictionary-vector multiplication. As the dictionary is structured here, it can be implemented in $\mathcal{O}(P\log(MN))$ using a filtering technique. In the optimal first-order gradient technique for sparse representations, which are known to be practically fast techniques for convex sparse approximation [2], the cost of each iteration is a constant time the complexity of each dictionary-vector multiplications. As the number of iterations for the convergence is generally not known and it is often much larger than K with coherent dictionaries, the overall algorithm is expensive for the real-time implementation. In the following, we will introduce two simple algorithms, which are actually modified versions of the corresponding sparse approximation methods, which are computationally cheap. Some theoretical guarantees for the touch location recovery using the following proposed method, is presented in [19]. These results are based upon bounding the error of recovering a set of touch locations in the neighbourhood of actual touch locations.

4.1 Support Constrained Positive Hard Thresholding (SCPHT)

This method is based on the hard-thresholding for sparse approximation [7, 15]. We first calculate $\mathbf{D}^T\mathbf{z}$ to find θ_t. We now start from the largest coefficient of θ_t, keep this coefficient and set the $R/2$-neighbourhood coefficients to zero. Note that SCPHT uses the values of coefficients and not on their *absolute magnitudes*, to consider the positivity of the sparse representation. $R/2$ has been chosen as the thresholding parameter, to consider the fact that the selected atoms can be away from the actual location of the closest touches. We repeat this process K times and let the remaining small coefficients be zero to derive θ. A pseudocode for this algorithm is presented in Algorithm 1. $\mathcal{H}_{R/2}$ thresholds all the coefficients in the $R/2$-neighbourhood of the index s_k, *i.e.* $\forall i,\ \|(\hat{\alpha}_i, \hat{\beta}_i) - (\hat{\alpha}_{s_k}, \hat{\beta}_{s_k})\|_2 \leq R/2$.
SCPHT includes one dictionary-vector multiplication, one sorting, which is $\mathcal{O}(P\log(P))$. The overall computational complexity is therefore $\mathcal{O}(P(\log(MN)+\log(P)))$, which is *slightly* more expensive, in order of magnitude, than a simple digital filtering.

4.2 Support Constrained Positive Matching Pursuit (SCPMP)

Matching Pursuit (MP) [13] is a greedy sparse approximation algorithm, which iteratively chooses a new non-zero components for θ, to reduce the approximation error. We restrict the set of possible non-zero elements in the atom selection step, *i.e.* to not have negative magnitude and not be close to one of the selected atoms, and introduce the modified algorithm of Algorithm 2. As this algorithm impose

Input: K, $\mathbf{D}$, $\mathbf{z}$ and R
Initialisation: $k = 1$, $0 \leq \lambda \ll 1$ and $\theta = \mathbf{0}$,
$\theta_t \leftarrow \mathbf{D}^T\mathbf{z}$
while $k \leq K$ and $\theta_t \neq \mathbf{0}$ **do**
  $s_k \leftarrow \operatorname{argmax}_s \theta_t(s)$
  $\theta(s_k) \leftarrow \theta_t(s_k)$
  $\theta_t \leftarrow \mathcal{H}_{R/2}(\theta_t, s_k)$
  $k \leftarrow k+1$
end while.

Algorithm 1: Support Constrained Positive Hard Thresholding

Input: K, $\mathbf{D}$, $\mathbf{z}$ and R
Initialisation: $k = 1$, $\theta = \mathbf{0}$ and $\mathbf{m} = \mathbf{1}_{P\times 1}$
$\mathbf{r} = \mathbf{z}$
$\theta_r \leftarrow \mathbf{D}^T\mathbf{r}$
while $k \leq K$, $\mathbf{m} \neq \mathbf{0}$ and $\max_s \mathbf{r}(s) > 0$ **do**
  $s_k \leftarrow \operatorname{argmax}_{s\in\operatorname{supp}(\mathbf{m})} \theta_r(s)$
  $\theta(s_k) \leftarrow \theta_r(s_k)$
  $\mathbf{m} \leftarrow \mathcal{H}_{R/2}(\mathbf{m}, s_k)$
  $\mathbf{r} \leftarrow \mathbf{r} - \mathbf{d}_{s_k}\theta(s_k)$
  $\theta_r \leftarrow \theta_r - \mathbf{D}^T\mathbf{d}_{s_k}\theta(s_k)$
  $k \leftarrow k+1$
end while.

Algorithm 2: Support Constrained Positive Matching Pursuit

some restrictions on the support, we call it here the Support Constrained Positive Matching Pursuit. $\operatorname{supp}(\mathbf{m})$ indicates the index set of non-zero elements of $\mathbf{m}$, $\mathbf{m}$ thus acts as a mask for the coefficient vector θ_r.
Although SCPMP is generally a computationally cheap structured sparse approximation algorithm, it is still a more computational demanding algorithm than SCPHT. In each step of SCPMP, we need to calculate a dictionary-vector multiplication, finding the maximum, updating r and updating θ_r[2] respectively with the complexities of order $P\log(MN)$, P, MN and P. Therefore, the total computational complexity of SCPMP with K iterations would be $\mathcal{O}\left(K\left(P\log(MN)+2P+MN\right)\right)$. The advantage of using SCPMP, instead of SCPHT, is demonstrated in the simulation section. SCPHT can only be competitive, when the magnitude of Gaussian functions, *i.e.* a_i's are roughly the same.

5 Simulations

In the first experiment, we show that SCPMP can actually improve the resolution of location recovery. We chose a 11 by 13 sensing grid and a complete and a 36 times overcomplete dictionaries, which were generated by regularly partitioning the touch sensor area. A five touches input signal was generated according to (1), where the location of touches were selected randomly while complying the minimum distance constraint, *i.e.* $R = 2\Delta$, and a_i's were ran-

[2] This step is computationally simplified using the precomputed Gramian matrix.

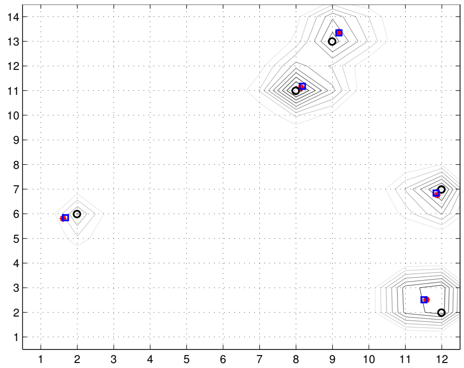

Figure 3: Five finger touches signal and the recovered touch locations. Stars, circles and squares respectively indicate the original locations, recovered by SCPMP using a complete and a 36 times overcomplete dictionaries.

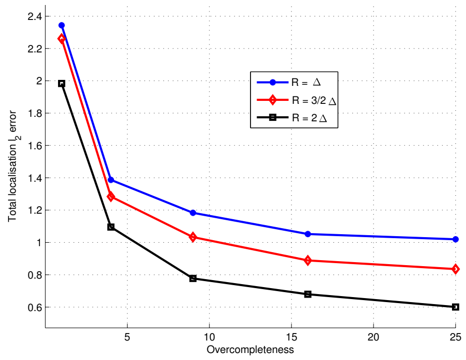

Figure 4: The average total reconstruction errors for different overcompleteness of regular (over-)complete dictionaries, over 1000 random trials.

domly selected between 0.5 and 1. We have plotted the original signal with some contour plots, the original locations of the touches by stars, the recovered locations using the complete dictionary by circles and the recovered locations using the 36 times overcomplete dictionary by squares in Figure 3. This figure shows that the proposed technique has recovered the closest points on the grid when the dictionary is complete and more accurate locations when the dictionary is overcomplete. If we repeat this simulation for different overcompleteness and measure the total ℓ_2 errors of the location recovery for 1000 randomly generated trials, we can quantify the success of sparse supper-resolution technique, which is shown in Figure 4. This figure shows the errors for three different closeness parameters R.

In the next experiment, we use SCPHT for the location recovery, using the setting of the previous experiment. We have plotted the average total error over 1000 trials in Figure 5, where $R = 2\Delta$ and a_i's were 1 or randomly selected between 0.5 and 1. When the magnitude of the touch signals can change, the performance of SCPHT is poor for overcomplete dictionaries, see the curve with circle indicators in this figure. However, when the magnitudes are roughly the same, here $a_i = 1, \forall i$, the error is reduced for the overcomplete dictionaries, see the curve with diamond indicators. It emphasises on the fact that for the magnitude sensitive touch sensors, we have to use a more complicated algorithm like SCPMP, as SCPHT fails in the super-

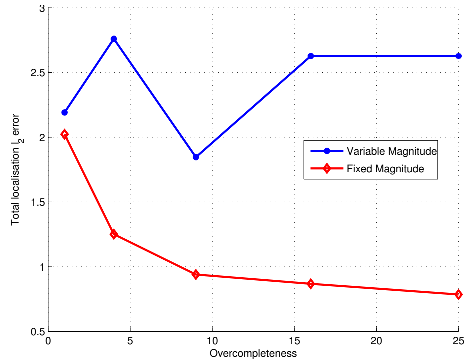

Figure 5: The average error using SCPHT over 1000 trials for fixed and variable touch magnitudes.

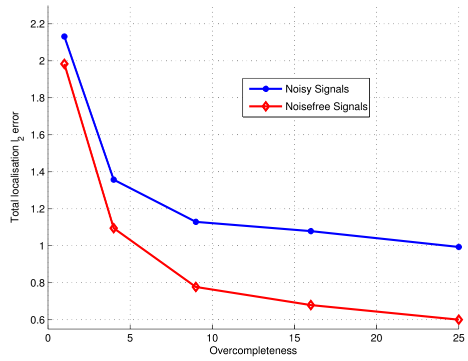

Figure 6: The location recovery error for the noisy and noisefree sensing signals.

resolution sensing here. However, for the capacitive touch sensors, the magnitude of each touch is due to the proximity of the finger, which makes a roughly constant magnitude for each touch.

In the last experiment, we investigate the noise sensitivity of the proposed super-resolution technique, using the SCPMP algorithm. If we add a zero-mean white noise, with standard deviation of 0.1, to the generated signal of previous experiments, when the magnitude of a_i is randomly between 0.5 to 1, we get the recovery errors which are plotted in Figure 6. The noiseless recovery result has also been plotted for the reference. This experiment demonstrates that the proposed super-resolution technique is fairly robust to the additive noise of sensors.

6 Summary

We formulated the super-resolution sparse touch sensing and presented two practical algorithms to improve the sensing resolution. The algorithms are the modified versions of the well-known Hard-Thresholding and Matching Pursuit sparse approximation algorithms. We showed that the touch sensing error reduces by the proposed techniques in the synthetic experiments. Although the resolution improvement here is significant, we may be able to even increase the resolution using a more efficient overcomplete dictionary, found by dictionary learning. We left this for the future work.

Acknowledgements

This work was partially supported by EU FP7, FET-Open grant number 225913 and EPSRC grants EP/J015180/1 and EP/K014277/1. MED acknowledges support of his position from the Scottish Funding Council and their support of the Joint Research Institute with the Heriot-Watt University as a component part of the Edinburgh Research Partnership in Engineering and Mathematics.

References

[1] R.G. Baraniuk, V. Cevher, M.F. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Transactions on Information Theory*, 56(4):1982 –2001, april 2010.

[2] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[3] James A Cadzow. Signal enhancement - a composite property mapping algorithm. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36(1):49–62, 1988.

[4] E. Candès and C. Fernandez-Granda. Towards a mathematical theory of super-resolution. *Communications on Pure and Applied Mathematics*, 2013.

[5] S. Chen and D. Donoho. Basis pursuit. In *Twenty-Eighth Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 41–44. IEEE, 1994.

[6] D. Donoho. Superresolution via sparsity constraints. *SIAM Journal on Mathematical Analysis*, 23(5):1309–1331, 1992.

[7] D.L. Donoho and J.M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.

[8] P.L. Dragotti, M. Vetterli, and T. Blu. Sampling moments and reconstructing signals of finite rate of innovation: Shannon meets strang–fix. *IEEE Transactions on Signal Processing*, 55(5):1741–1757, 2007.

[9] J. J. Fuchs. On sparse representations in arbitrary redundant bases. *IEEE Transactions on Information Theory*, 50(6):1341–1344, 2004.

[10] H. Hou and H. Andrews. Cubic splines for image interpolation and digital filtering. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 26(6):508–517, 1978.

[11] C. Luo, M.A. Borkar, A.J. Redfern, and J.H. McClellan. Compressive sensing for sparse touch detection on capacitive touch screens. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems,*, 2(3):639–648, 2012.

[12] C. Luo, J McClellan, M. Borkar, and A. Redfern. Sparse touch sensing for capacitive touch screens. In *International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 2545–2548, 2012.

[13] S. Mallat and Z. Zhang. Matching pursuits with time frequency dictionaries. *IEEE Trans. on Signal Processing*, 41(12):3397–3415, 1993.

[14] T. O'Connor. mTouch^{TM} projected capacitive touch screen sensing theory of operation. Technical Report TB3064, Microchip Technology Inc, 2010.

[15] K. Schnass and P. Vandergheynst. Average performance analysis for thresholding. *IEEE Signal Processing Letters*, 14(11):828–831, 2007.

[16] J. Schöning, P. Brandl, F. Daiber, F. Echtler, O. Hilliges, J. Hook, M. Löchtefeld, N. Motamedi, L. Muller, P. Olivier, T. Roth, and U. von Zadow. Multi-touch surfaces: A technical guide. Technical Report TUM-I0833, Institute for Geoinformatics University of Münster, 2008.

[17] J.A. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Trans. on Information Theory*, 50(10):2231–2242, 2004.

[18] M. Vetterli, P. Marziliano, and T. Blu. Sampling signals with finite rate of innovation. *IEEE Transactions on Signal Processing*, 50(6):1417–1428, 2002.

[19] M Yaghoobi, S. McLaughlin, and M.E. Davies. Super-resolution sparse multitouch sensing. in preparation.

[20] J. Yang, J. Wright, T.S. Huang, and Y. Ma. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, 19(11):2861–2873, 2010.