An Educational Review of Compressed Sensing

Sven Dorkenwald
ziti, Heidelberg
Email: sven.dorkenwald@googlemail.com

***Abstract*—Compressed Sensing (CS, also referred to as Compressive Sensing) is a recent signal processing technology for efficient signal acquisition and reconstruction. CS exploits the fact that most signals are sparse in some domain, usually not the sensing domain, and tries to capture a signal directly in a compressed manner. It can reduce the number of measurements in many settings drastically compared to the standard approach - Nyquist-Shannon (NS) sampling. Here, I attempt an educational review of CS, intended to give an easy entry to the matter. Besides the theory behind CS, I present own experiments and two examples of applications of CS.**

I. Introduction

Data acquisition aims to capture signals accurately, without loosing information. Nyqusit and Shannon showed that this is possible when sampling the signal with at least twice the highest frequency present (Nyquist Shannon sampling theorem). However, this is often followed by a compression step before storing or sending it to a receiver. Hence, it seems possible to store the same information captured by the sampling process in a smaller (compressed) representation.

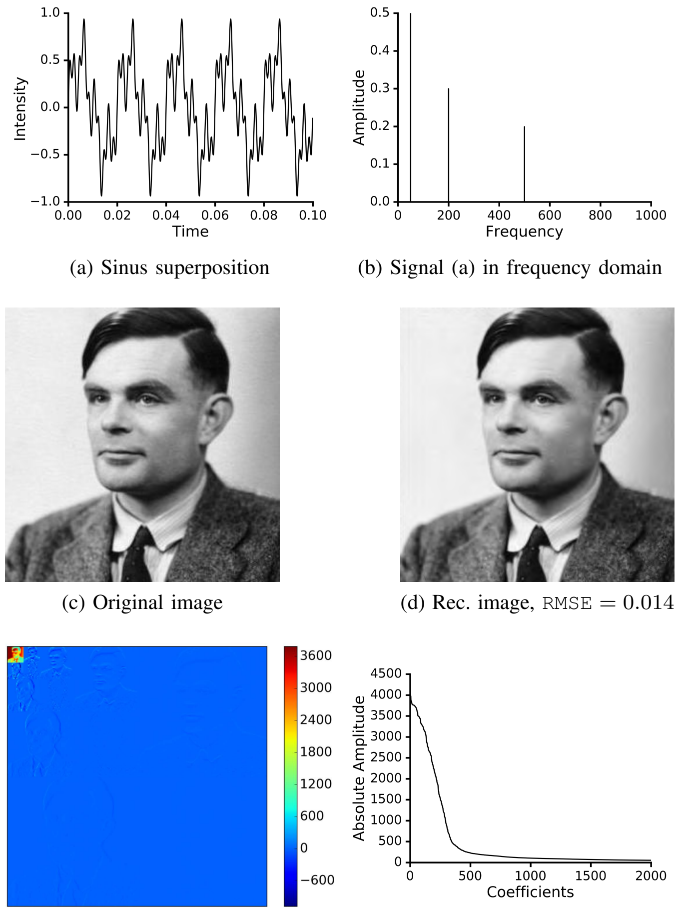
Oversampling has disadvantages. In many cases, sensing is costly (e.g. imaging in low light or in the infrared spectrum) and in others, sensing time is critical or even limited (e.g. medical applications, such as magnetic resonance imaging (MRI)). Furthermore, predominant compression algorithms are designed to shift the cost of compression to the recorder side to optimize decompression. However, in cases with limited compute capabilities on the recorder side, this can be problematic (e.g. a satellite taking and transmitting images).

CS [1]–[3] addresses these problems for many settings. Effectively, CS is able to capture the signal in the domain in which the signal is sparse, while being agnostic to the specific domain itself. However, CS needs to be incorporated in the setup and poses trade-offs as I will discuss in the course of this work.

This review is, however, not intended to be exhaustive, but rather introductory. Further, I am going to show experiments that I performed, namely standard CS image sensing and recovery of a corrupted image. Lastly, I present to applications of CS that are exemplary (single pixel camera) and already established (MRI).

To assess the quality of reconstructions presented in this work I am going to use the root mean squared error (RMSE). The RMSE for a reconstruction x_r with regard to its original counterpart x_o is defined as:

$$\mathtt{RMSE}(x_r) = \sqrt{(x_o - x_r)^2} \tag{1}$$



(a) Sinus superposition
(b) Signal (a) in frequency domain
(c) Original image
(d) Rec. image, $\mathtt{RMSE} = 0.014$
(e) Wavelet representation
(f) Absolute coefficients

Fig. 1. Examples for sparsity in real data.

II. Sparsity - why we can compress

The fundamental reason why compressed sensing works is the same why any signal compression algorithm works: sparsity. A signal is sparse when only a small fraction of its coefficients is nonzero. Signals might however not be sparse in the domain in which they are sensed. Hence, many compression algorithms such as JPEG [4] use basis transformations (e.g. Fast-Fourier (FFT), Discrete Cosinus (DCT), Wavelet (WT)) to transform signals in a domain in which they are sparse.

Fig. 1a and Fig. 1b illustrate this for a superposition of three sinus waves of different frequencies. While the original signal would have to be sampled with 1 kHz (the highest frequency

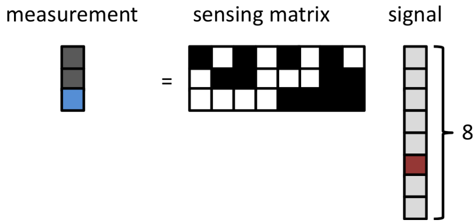


Fig. 2. One solution to the counterfeit example presented as dot product. Each row of the sensing matrix is a single weighting with black boxes decoding which coins are included. The eight coins are summarized as signal and the measurement is the result of the three weightings. The red box represents the counterfeit and the blue box an unexpected weighting. E.g. the first measurement includes coins 1, 3, 5 and 7. There are multiple solutions to this problem.

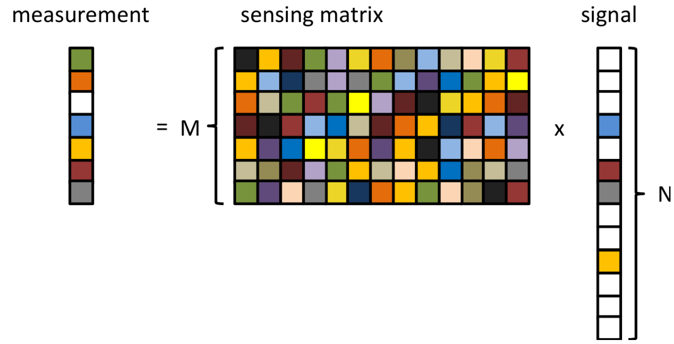


Fig. 3. CS data acquisition of a k-sparse signal with a random sensing matrix illustrated as dot product.

is 500 Hz), only nine parameters are needed to describe the signal completely (3 frequencies, 3 factors, 3 phases), which can be extracted by applying an FFT to the signal.

Another example is shown Fig. 1c and Fig. 1d. The original image has 82944 parameters (pixel gray values), but when transformed with a WT (Fig. 1e) most parameters become very small (Fig. 1f). When setting the 90% smallest coefficients to 0, the reconstruction (transforming back with the inverse WT) bares very little difference to the original image (Fig. 1d).

Most compression algorithms follow a more sophisticated approach than simply erasing the n% smallest coefficients. They also have to deal with storing the position of the remaining ones. This is for instance accomplished by using specific patterns that implicitly store the position. Nonetheless, at their core, these algorithms use the fact that most real-world signals are sparse, and thereby compressible, in some domain. CS makes use of this by assuming that the sensed signal is sparse in some domain, making a compressed acquisition possible.

III. Theory

A. Counterfeit example

To get a better intuition for the concepts presented in this section, I will start with a short thought experiment [5]:

We have eight coins of which exactly one is a counterfeit. The counterfeit looks the same as the others, but it has a different weight. How many measurements with a scale does it take to always identify the counterfeit?

This could be solved with seven measurements by weighing one coin after another, but then this does not leverage the implicit information that we have about the coins: only one coin is a counterfeit. In other words: the signal is sparse. In fact, one would only need three digits to exactly describe the set of coins by using a binary basis for the position of the counterfeit. We could also call this a compressed representation of our coin situation.

To make use of this, one must measure multiple coins at once in such a way that the results of all measurements can only be explained with one solution to the task. An example of a set of combinations is shown in Fig. 2 represented as an inner matrix product. For instance, in the given setting one would know that the first coin is the counterfeit if only the first weighting has an unexpected weight.

This algorithmic correctness independent of which coin is the counterfeit and is thereby agnostic to the exact signal as long as the degree of sparsity stays the same. As we will see later, CS has very similar properties.

B. Sensing sparse signals

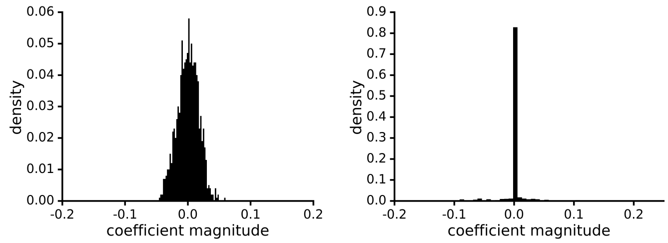
As described earlier, most signals are sparse in some domain, but this is often not the sensing domain. For the purpose of explaining CS, I first will assume that we are receiving a sparse signal and will make necessary adjustments later.

Suppose we are sensing a sparse signal x with N parameters of which only k are nonzero. Such signals are then called k-sparse. I will represent x as a vector of length N (signal in Fig. 3), which does not mean that the signal has to be 1D-dimensional. One could think of x as the flattened version of any signal one might sense. It is also worth noting that even if we know that the signal is k-sparse we do not know which of the N entries are nonzero.

Sensing x the classical way would mean to measure each of its entries one by one. This can be thought of multiplying x with the identity matrix, which would give the sensing matrix Φ. Although this captures all information in the signal, it is apparent that we oversampled the signal since k measurements would have been enough to accomplish the same result.

Hence, having a sensing matrix Φ with dimensions $M{\times}N$ where M is close to k is desirable. This matrix could not have full rank, because $M < N$. The rank of a matrix is defined as the minimal number of linear independent rows or columns in a matrix. This reflects that many arbitrary x vectors will be projected to the same vector and we would loose the information of which was which.

Here we can make use of the knowledge that x is not an arbitrary signal, but a sparse one. Hence, the inner product will be essentially a product of a $M{\times}k$ submatrix with the nonzero entries in x (Fig. 3), which in turn can have full rank. However, building a sensing matrix Φ with the guarantee that any such submatrix has full rank is NP-hard and therefore



(a) parameters with L2-regularization (b) parameters with L2-regularization

Fig. 4. Distribution of parameters for a regression with L2- and L1-regularization. Parameters for Φ and y were sampled from an iid. Gaussian distribution with $M = 200$ and $N = 1000$. This example was inspired by [8].

not applicable. This problem is called the Restricted Isometry Problem (RIP).

It has been shown that the RIP is fulfilled to a very high likelihood when sampling the parameters of Φ from an independent and identically distributed (i.i.d.) Gaussian distribution [6], [7] and choosing M large enough. Mathematically it can be shown that $M > O(K \log(N/k))$, and thereby only grows with the log of the data size. In experiments M is often found empirically, because one also has to account for not knowing k exactly.

C. Recovering sparse signals

Now that we have found a way to sense x without formally losing information given that the signal is k sparse, we have to recover x from its measurement y. I will use $\hat{x}$ as the recovered signal from y. This poses a regression problem in the classical sense:

$$min(||\Phi\hat{x} - y||_2) \tag{2}$$

with the Lp-norm $|| \cdot ||_p$ defined as

$$||a||_p = \big(\sum_i |a_i|^p\big)^{\frac{1}{p}} \tag{3}$$

This could be solved with a standard method like least-squares (LS), ordinary LS will find a $\hat{x}$ that solves this system of linear equations perfectly. In fact, there are infinitely many solutions to this system, because it is underdetermined ($M < N$). However, we are not interested in any arbitrary solution because we want $\hat{x}$ to be as close to x as possible.

Again, we have to make use of the fact that our desired signal is sparse. The standard way in Machine Learning (ML) to restrict the size of $\hat{x}$ is to add a term that penalizes $||\hat{x}||_2$ and is called regularization. Adding this to the minimization problem leads to

$$min(||\Phi\hat{x} - y||_2 + \lambda||\hat{x}||_2) \tag{4}$$

with the regularization parameter λ to control the strength of the regularization. While this leads to a more robust solution for the linear system, it does not induce sparseness. Instead of the size of $\hat{x}$ we would actually like to penalize the number of nonzero entries, which effectively uses the L0-norm for the regularization term. Using the L0-norm however would make the problem non-convex and therefore much harder to solve. The solution to this is to use the L1-norm instead. This seemingly small change is a key pillar of CS. The optimization problem is then

$$min(||\Phi\hat{x} - y||_2 + \lambda||\hat{x}||_1) \tag{5}$$

The difference between using the L1-norm compared to the L2-norm is illustrated in Fig. 4. Here, I randomly chose Φ and y and solved the linear system with an L1- and L2-regularization. It can be seen that the parameters in the L2 case tend to be small, but none of them are exactly zero. However, in the L1-case over 80% of the parameters are actually zero, meaning that the recovered signal is sparse.

D. Generalization to non-sparse signals

So far, we assumed that our signal x is canonically sparse. In general, x is not sparse in the sensing domain, but as discussed earlier, there likely is a basis in which x is sparse. With such a basis Ψ and x' as the representation of x in that basis we get the simple relationship

$$\Psi x' = x \tag{6}$$

changing our sensing to

$$y = \Phi\Psi x \tag{7}$$

At first, it seems as if our sensing process is dependent on our basis Ψ. However, we can use that our transformation matrix Ψ is always an orthogonal matrix and the product of an i.i.d. Gaussian matrix and an orthogonal matrix results in an i.i.d. Gaussian matrix Φ' [9]. Hence, we can just assume that our sensing matrix inhibits the transformation matrix, so we directly sense with Φ'.

In the recovery process, we can now try any transformation matrix. We get

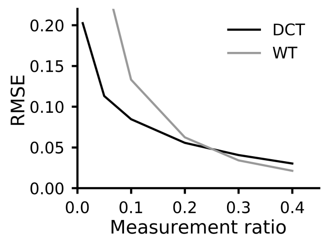
$$min(||\Phi'\Psi\hat{x}' - y||_2 + \lambda||\hat{x}'||_1) \tag{8}$$

as our minimization problem because we still want to recover $\hat{x}$ in the domain in which it is sparse. We get $\hat{x}$ by transforming back from the sparse domain in the sensing domain.

E. Theory summary

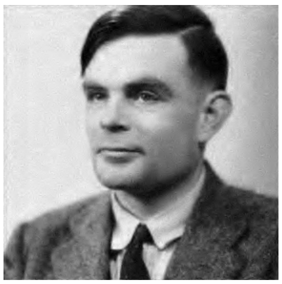
CS senses the signal with M random measurements, where M is found empirically and depends on the sparsity of the signal. The signal is reconstructed with an L1-regularized regression to enforce sparsity in the recovered signal. The domain in which the signal is sparse does not have to be known while sensing, and multiple transformations can be tested for the reconstruction.

Thereby, CS addresses the problems of NS sampling described earlier. It needs fewer measurements and shifts the computationally intensive part to the receiver. Hence, CS is especially useful in settings where the dot product is embedded in the measurement.

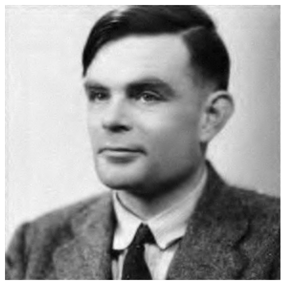


(a) RMSE for different ratios of measurements to image size ($\frac{k}{x \cdot y}$)

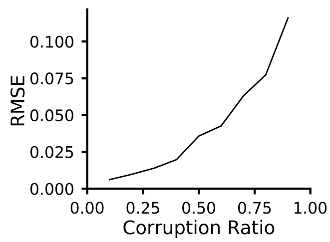
(b) DCT, 10%, $\mathtt{RMSE} = 0.085$



(c) WT, 30%, $\mathtt{RMSE} = 0.034$



(d) WT, 40%, $\mathtt{RMSE} = 0.021$



(e) RMSE for different cor. ratios

(f) 40% cor., $\mathtt{RMSE} = 0.020$

Fig. 5. CS experiments. Standard CS for an image using WT and DCT basis (a-d) and image recovery with WT basis (e, f).

IV. Experiments

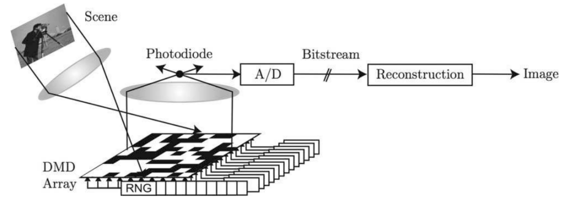
I conducted my own experiments and open-sourced the `python` code under `https://github.com/sdorkenw/CompressedSensing`. For regression I relied on the `Lasso` implementation of `scikit-learn` [10]. For WTs I used `pywt` and built a 2d variant producing images such as the one shown in Fig. 1e. 2D DCTs were implemented by chaining 1D DCTs from `scipy's fftpack` [11] using

$$\mathtt{DCT}_{\mathtt{2D}} = \mathtt{DCT}_{\mathtt{1D}}(\mathtt{DCT}_{\mathtt{1D}}(\mathtt{img}^T)^T) \quad (9)$$

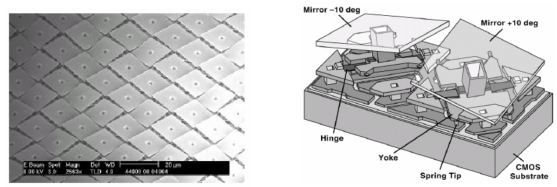
for an image `img`.

A. Classical compressed sensing for images

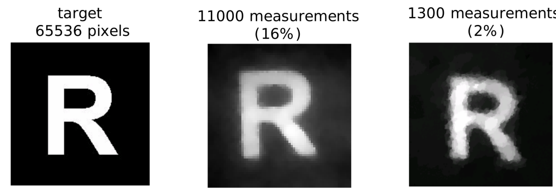
Following the classical procedure of compressed sensing I decreased the measurements k for the image shown in Fig. 1c. Using the RMSE as a measure for reconstruction quality I optimized the regularization parameter (Eq. 8) and plotted RMSE against the ratio $\frac{k}{x \cdot y}$ for both the WT and the



(a) Experiment scheme



(b) Close up of pixel array



(c) Reconstruction of a sample image

Fig. 6. Single pixel camera setup and example measurements. Images taken from [12].

DCT basis (Fig. 5a).
Reconstructions with more measurements tend to be better with WTs than with DCTs. However, visually good measurements were only reached for ratios above 30% (Fig. 5b, 5c, 5d). These are still worse than the compressed image shown in Fig. 1e where only 10% of the coefficients were used.

B. Image recovery

An alternative usage for CS is the recovery of corrupted signal. In this case I assumed that the corrupted pixels of an image are known and used CS to recover these pixels. The locations of the corrupted pixels can then be seen as the sensing matrix with binary entries and the recovery then follows the standard CS approach. Fig. 5e shows the RMSE for different corruption ratios and the recovered image at 40% corruption is presented in Fig. 5f.
As expected, recovery gets increasingly harder for higher corruption ratios, but RMSEs for higher corruption rates are still far below the RMSE of a random noise ($\mathtt{RMSE} = 0.423$ compared to target image).

V. Applications

A. Single pixel camera (SPC)

The SPC [13] represents a straight forward implementation of CS in hardware. Light from an image is focused on a mirror array reflecting the light onto a photodiode. Each mirror can be set to an angle of +10 or -10 degrees, translating to reflecting the light onto the diode or not. The photodiode

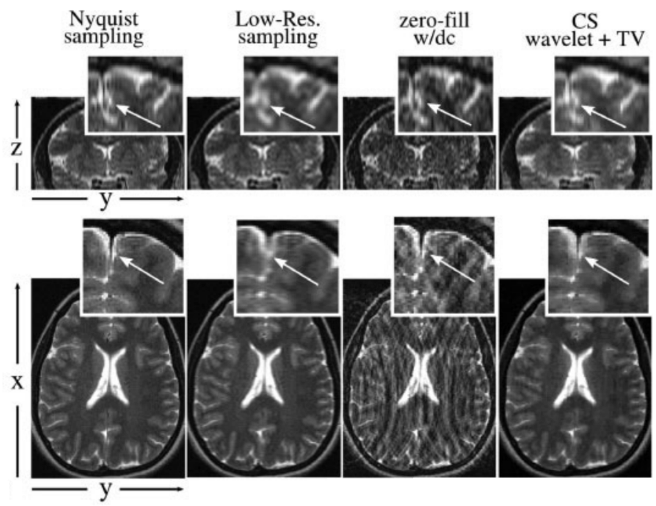


Fig. 7. Comparison of Nyqusit sampling with CS applied to MRI at 2.4 accelleration (2008). CS performs visually equal to Nyquist sampling and is superior to low-res. sampling at the same rate or zero-filling Fourier reconstruction applied to low-res. sampling. Image taken from [15].

measures the sum of the light coming from a specific mirror array constellation. One mirror configuration represents one row in the sensing matrix and measuring the light from many configurations is then the dot product from the incoming signal with the sensing matrix.

While the SPC seems almost pointless to capture images in the domain of visible light, it has been successfully used in regimes where imaging is more expensive, such as low light and infrared [13]. Imaging visible light is cheap, because CCDs and CMOS sensors have become cheap. However, photomultiplier and infrared diodes are expensive and building large arrays with them is very costly. Here, the SPC only needs one of these building blocks to operate. Current development of the SPC for instance lead to a lenseless camera [14].

B. Magnetic resonance imaging (MRI)

MRI is one of the fields in which CS had the biggest impact on a wider scale and already made it into the maturity phase [15], [16]. MRI attempts to image hydrogen atoms - mainly water - in a patient. A large number of slices are taken in different angles with the aim to recover the 3D image of a specific section. Early, but impressive, results of applying CS to MRI are presented in Fig. 7. For more recent work see for instance [17], [18].

On the one side, CS fits nicely in the repetitive acquisition process of MRI and is able to reduce data acquisition many fold without sacrificing resolution. On the other side, some MRI applications are time critical like the Cardiac Cine (imaging of heart pump cycle) and require repeated breath holding of the patient. To address this Siemens launched a CS backed device in 2016 that is able to image Cardiac Cine during normal breathing [19], [20]. This application makes additional use the high sparsity in the time domain to speed up the acquisition over ten fold.

Additionally, the trade-off between quality and number of measurements in CS as shown in Fig. 5a allows for an easy adjustment of the acquisition quality and speed for the specific situation.

VI. Related work

CS has many similarities to methods in ML and statistics such as non-negative matrix factorization [21] and autoencoders [22]. In matrix factorization, the compressed representation (in CS: the measurement) is thought of as generalization of the original, often incomplete, data from which predictions can be made for. For example, this was used by the winner of the Netflix challenge [23], where the task was to infer movie preferences of users based on their ratings and ratings of other users. Autoencoders, and especially variational autoencoders, [24] can be used to build generative models. An exemplary one is [24].

It is not surprising that CS is blending in with these methods and various papers have been proposed that for instance combine matrix factorization and CS [25] or replace the L1-regularization with a generative model [26].

VII. Conclusion

In this work, I presented a high-level introduction to CS compiled from more in-depth reviews [5], [27], [28] and evaluated experiments and applications of CS.

CS is an eruptive technology that is able to change the way of acquisition in many fields. Since CS is still young, it has not left the research stage in many areas, but for instance in MRI it already made it into industrial production.

While CS requires more computational resources and development, it has many advantages over standard NS sampling and compression. For one, it is able to speed up acquisition oftentimes many-fold. It shifts the computational intensive part to the decoder. It is agnostic to the basis in which the signal is sparse, meaning that many different basis can be tested when decoding. And finally, it poses an interesting trade-off between signal quality and acquisition speed that can be adjusted to the specific situation.

The performed experiments are only a glimpse of what CS can achieve. State-of-the-art algorithms would outperform them easily. Today, better solvers are getting developed that use better priors than the pure sparsity of the reconstructed signal by using for instance generative models [26]. Especially in restricted applications such as MRI, stronger priors can help to make recovery easier, leading to even fewer measurements needed.

References

[1] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[2] E. J. Candes and T. Tao, "Near-Optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[3] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[4] G. K. Wallace, "The JPEG still picture compression standard," *Commun. ACM*, vol. 34, no. 4, pp. 30–44, Apr. 1991.

[5] K. Bryan and T. Leise, "Making do with less: An introduction to compressed sensing," *SIAM Rev.*, vol. 55, no. 3, pp. 547–566, 2013.

[6] D. Achlioptas, "Database-friendly random projections," in *Proceedings of the Twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, ser. PODS '01. New York, NY, USA: ACM, 2001, pp. 274–281.

[7] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," *Constr. Approx.*, vol. 28, no. 3, pp. 253–263, 1 Dec. 2008.

[8] J. Nathan Kutz, *Data-Driven Modeling & Scientific Computation: Methods for Complex Systems & Big Data*. OUP Oxford, 8 Aug. 2013.

[9] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Probl.*, vol. 23, no. 3, pp. 969–985, Jun. 2007.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay, "Scikit-learn: Machine learning in python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Nov. 2011.

[11] E. Jones, T. Oliphant, and P. Peterson, "SciPy: Open source scientific tools for python," 2001.

[12] R. Baranuik and K. Kelly, "Lecture: A Single-Pixel compressive camera," 2007.

[13] M. F. Duarte, M. A. Davenport, D. Takbar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-Pixel imaging via compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 83–91, Mar. 2008.

[14] M. S. Asif, A. Ayremlou, A. Veeraraghavan, R. Baraniuk, and A. Sankaranarayanan, "FlatCam: Replacing lenses with masks and computation," in *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, Dec. 2015, pp. 663–666.

[15] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 72–82, Mar. 2008.

[16] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn. Reson. Med.*, vol. 58, no. 6, pp. 1182–1195, Dec. 2007.

[17] R. Otazo, D. Kim, L. Axel, and D. K. Sodickson, "Combination of compressed sensing and parallel imaging for highly accelerated first-pass cardiac perfusion MRI," *Magn. Reson. Med.*, vol. 64, no. 3, pp. 767–776, 2010.

[18] L. Feng, L. Axel, H. Chandarana, K. T. Block, D. K. Sodickson, and R. Otazo, "XD-GRASP: Golden-angle radial MRI with reconstruction of extra motion-state dimensions using compressed sensing," *Magn. Reson. Med.*, vol. 75, no. 2, pp. 775–788, Feb. 2016.

[19] Siemens, "FDA clears compressed sensing MRI acceleration technology from siemens healthineers," https://usa.healthcare.siemens.com/press/pressreleases/healthcare-news-2017-02-21-1, 21 Feb. 2017, accessed: 2017-3-13.

[20] T. Kido, T. Kido, M. Nakamura, K. Watanabe, M. Schmidt, C. Forman, and T. Mochizuki, "Compressed sensing real-time cine cardiovascular magnetic resonance: accurate assessment of left ventricular function in a single-breath-hold," *J. Cardiovasc. Magn. Reson.*, vol. 18, no. 1, p. 50, 2016.

[21] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 21 Oct. 1999.

[22] P. Baldi, "Autoencoders, unsupervised learning, and deep architectures," *ICML unsupervised and transfer learning*, 2012.

[23] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[24] D. P. Kingma and M. Welling, "Auto-Encoding variational bayes," 20 Dec. 2013.

[25] S. C. Jiho Yoo, "Matrix co-factorization on compressed sensing," in *In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI*, 2011.

[26] A. Bora, A. Jalal, E. Price, and A. G. Dimakis, "Compressed sensing using generative models," 9 Mar. 2017.

[27] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[28] M. Fornasier and H. Rauhut, "Compressive sensing," in *Handbook of Mathematical Methods in Imaging*, O. Scherzer, Ed. Springer New York, 2011, pp. 187–228.