Compressive sensing for reconstruction of 3D point clouds in smart systems

Ivo Stančić
Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture
University of Split
Split, Croatia
istancic@fesb.hr

Miloš Brajović, Irena Orović
Faculty of Electrical Engineering
University of Montenegro
Podgorica, Montenegro
milosb@ac.me, irenao@ac.me

Josip Musić
Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture
Split, Croatia
jmusic@fesb.hr

***Abstract*—Performing an accurate 3D surface scan of everyday objects is sometimes difficult to achieve. Using the 3D scanner as a main sensor in a fast-moving mobile robot emphasizes this issue even further. When small robots with limited payload are considered, the professional Lidar systems are not likely to be embedded due to their weight, dimensions and/or high cost. Introduction of simple structured-light scanners makes possible fast scanning, effective robot detection and evasion of obstacles. Nevertheless, some obstacles may still be difficult to detect and recognize, primarily due to limitations of scanner's hardware which results in a low number of reconstructed surface points. In this paper a compressed sensing technique, primarily used for the reconstruction of 2D images, is utilized to enhance the quality of 3D scan, by increasing the number of reconstructed 3D points to the scanner's theoretical maximum. Obtained results demonstrated the feasibility of the approach in terms of mean square error.**

I. Introduction

The basic requirement for any mobile robot is an accurate knowledge of its surroundings, in order to localize itself in the space, avoid obstacles in its path, and demonstrate a smart behavior in general. This leads to the introduction of robot perception systems, whose main function is to detect the surrounding objects and avoid possible collisions. The detection and recognition of potential obstacles, especially in the dynamic surroundings is a challenging task, and a simple detection of obstacles (without an actual recognition) in the robot's vicinity is sometimes insufficient.

In the search for a solution, researchers and engineers have introduced several sensor systems, which are based on optoelectronics. They utilize simple geometry principles [1]-[2], with the emphasis on multi-camera systems and various 3D scanners. The basic principle is similar to the function of human vision [1]. Downside of this approach is the introduction of complex calculations, which are often difficult to execute in real-time with robot's on-board computer.

In contrast to the camera systems, the laser scanners offer a simpler calculation of the object's location in the space, where a laser point (line) is moved by a precise electromechanical system (the rotating mirror in most systems). Since the number of points returned is relatively low, the detection based solely on laser data may be unreliable [3]. The industry-grade laser scanners (e.g. SICK) [4] are in some cases more expensive than the robot itself, or too bulky to be efficiently utilized.

Structured light scanners were introduced as a low cost alternative to Lidar scanners [5]-[6], since they use a simple light pattern projector and off the shelf components. The development of the Microsoft Kinect system, which is in fact a more advanced structured light sensor, inspired researchers to use it as a robot navigation device. Correa et. al. [7] have successfully implemented the Microsoft Kinect as a sensor for indoor navigation of autonomous surveillance mobile robot.

Both structured light scanners and laser scanners are easier to implement than a multi-camera system; nevertheless they inherit some of the common imperfection of the optoelectronic system. The challenging surface patterns like the camouflaged objects, object with a low reflective ratio, parts of the object in the shade and the object lighted by other light source may be difficult to reconstruct due to a low number of 3D points in the resulting point cloud. Each scan can retrieve a fixed (maximum) number of reconstructed points (one projection of the light structure, or a full Lidar swipe). Techniques used in the compressed sensing, which are primarily developed for the reparations of damaged/undersampled 2D images, may be used for increasing the quality of a 3D scan, and supplement the missing 3D points in a point cloud. A similar approach was already successfully used for the refinement of depth-maps obtained by the Microsoft Kinect sensor [8].

Sparse signal processing and the compressed sensing (CS) attracted a significant research interest during the last decade [9]-[23]. CS deals with the reconstruction of randomly undersampled signals with the assumption that these are sparse in a known transformation domain. The reduced set of measurements often represents a consequence of a sampling strategy, in order to reduce the data size requirements and the number of acquisitions, preserving the same quality of the information as if these values are available [14]-[19]. On the other hand, known denoising techniques from the robust theory, such as the L-statistics, are used to eliminate signal samples corrupted by high noise [13], [20], [21]. Due to the random nature of the noise, the corrupted values assume random positions and if the sparsity condition is satisfied, CS reconstruction algorithms can be applied in the reconstruction of the eliminated values.

The theoretical foundation of CS lies in fact that the missing samples can be reconstructed by solving an undetermined system of linear equations with the additional sparsity constraint [9]-[12], [14]-[19]. Hence, an adequate measure of sparsity is exploited in the reconstruction procedures. A natural way to measure the sparsity is the so-called ℓ_0-norm, that is, the number of non-zero signal coefficients in the observed sparse transform domain.

It is crucial to emphasize that direct variations of unavailable samples values measuring ℓ_0-norm at the same time is an NP (non-deterministic polynomial-time) hard problem. Linear programming techniques and gradient-based algorithms are applied in the reconstruction by relaxing the sparsity constraint involving the ℓ_1-norm. Many studies [9]-[12] have confirmed that in the domain of interest this relaxation procedure is adequate in the CS context. Several reconstruction procedures are based on this relaxation [14]-[19]: well-known convex optimization algorithms such as primal-dual interior point methods and gradient-based methods (Orthogonal Matching Pursuit (OMP), Gradient Pursuit and CoSaMP).

In this paper, for the purpose of the observed 2D reconstruction problem, the gradient-based reconstruction algorithm presented in [13] is applied. The adaptations of the algorithm for the case of 2D signals and transforms are presented in [22]. However, in this work the reconstruction of missing values of disparity matrix (and not the pixel value based map) is based on the variation of their values in a steepest descent manner, minimizing the ℓ_1-norm-based sparsity measure. The available samples are fixed during the reconstruction, thus setting the constraints for the sparsity minimization.

II. Theoretical Background

A. 3D scanner system

The method evaluated in this paper is presented as an improvement of an in-house built structured light scanner, with the current accuracy in the 3rd dimension of 1.52 mm and RMSE below 1 cm [5]. It is built from off-the shelf components, which include a DLP (Digital Light Processing) projector, a digital camera and a computer. The design is based on a stereovision system [6], where DLP projector acts as an active component and camera is the passive one. If a single light ray is projected from a projector (noted as B on Fig. 1) it passes through the projector frame (noted as point B') and hits the target in point C. The reflected light ray is captured by camera through its plane (noted as point A'). The pixel on camera plane (A') and the pixel on projector plane (B') positions depend on an angle between the camera and the object (α) and the angle between the projector and the object (β). If the exact positions of the camera and the projector in the world coordinate frame are known, the problem of reconstructing exact position of point C is reduced to a triangulation problem. The position of point C in the reference coordinate frame is derived using equations (1) and (2), where A' and B' are its coordinates in the camera and projector planes, respectively; P_c and P_p are the camera and projector matrices; τ is a triangulation function; and H is the linear transformation that transforms *A'= HB'* [1].

$$C = \tau\left(A', B', P_c, P_p\right) \tag{1}$$

$$\tau = H^{-1}\left(A', B', P_c H^{-1}, P_p H^{-1}\right) \tag{2}$$

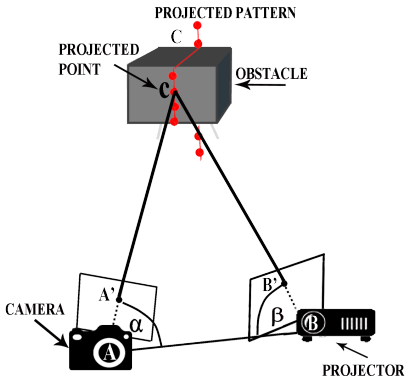

Figure 1. Triangulation of a surface point

If the vertical line is projected, and there is an object present in the front of the scanner system, line is curved on the camera side (Fig. 2). This is basic idea how the proposed scanner system operates.

By placing a plane at infinity distance from the scanner, the projected pattern (matrix containing 41 x 41 points) captured by the camera is a template image for objects at the infinity. By placing any object between a plane at infinity and scanner system, similar image of the projected matrix is captured but with projected points horizontally shifted towards the projector side, where the disparity (horizontal shift) is in direct function of distance between the scanner and the object (Fig. 2). Placing a plane at infinity distance is simple to achieve in a simulated scanner, while in real-life scenario, for obvious reasons, plane should be placed at scanner's maximum operating distance.

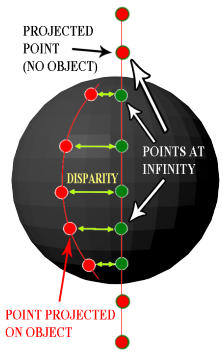

Figure 2. Disparity as seen from camera, as a result of an object placed in front of the scanner system

The horizontal displacement of each projected point with index (*i,j*) is stored as a value of the matrix at position (*i,j*).

The complex 3D reconstruction problem is in this part reduced to a 2D problem, which can be represented with a 2D image with rows/columns that directly correspond to rows/columns of the projected matrix, with values that correspond to the disparity at a given 2D location. In real scenario the obtained 2D matrix may have some missing elements, due to fact that some complex shapes can hide projected points from the camera.

B. Scanner simulator

The simulator implemented in this paper is mimicking a real-life scanner [5]. The virtual scanner is used rather than a real scanner in this part of the development, as it allows a fully controllable scene, which is hard to achieve even in the laboratory conditions. The simulated virtual scanner is created in Blender environment, and consist of a camera and a light ray projector. The objects of known characteristics and dimensions are placed in front of the scanner system in the virtual scene. The result of Blender simulation is a video stream recorded from the simulated scanner's camera, which has similar characteristics as a real camera. The projected pattern consist of 41 x 41 points in the matrix format, where each point is projected at a single frame, thus eliminating the possible error due to the point missclassification. In the process the frame rate (of virtual camera) was set to 30 Hz.

The simulated scanner is limited to 41 x 41 projected points, due to the fact that projected light ray object (cone) in Blender can only accept rounded numbers as angle of a cone tip, and the smallest possible angle (1 degree) was chosen. A higher density of points would be possible in a real-life projector, which is limited only by the projector resolution. Besides beneficial increase of scanners resolution, the real-life scanner would introduce some hard-to control effects, like unknown camera and projector lens distortion, errors due to small vibrations and effects of external light sources. For those reasons, effectiveness of the proposed algorithm is testes solely on a simulated scene.

Algorithm that calculates 3D locations of projected points is using previously obtained video stream [5]. Whole algorithm is fully implemented in Matlab environment, all data analysis and results visualizations are also done using Matlab tools.

III. The Compressed Sensing and the Disparity Matrix

A. Modelling the disparity matrix

Let us observe the disparity matrix $f(n,m)$ of size $M \times M$. The 2D DCT (Discrete Cosine Transform) of the considered disparity matrix has the following form (DCT II) :

$$C(k_1,k_2) = \sum_{n=0}^{M-1}\sum_{m=0}^{M-1} a_{k_1}a_{k_2} f(n,m) b_{n,m}(k_1,k_2). \quad (3)$$

The corresponding inverse transform is defined by:

$$f(n,m) = \sum_{k_1=0}^{M-1}\sum_{k_2=0}^{M-1} a_{k_1}a_{k_2} C(k_1,k_2) b_{m,n}(k_1,k_2), \quad (4)$$

with

$$b_{n,m}(k_1,k_2) = \cos\left(\frac{2\pi(2n+1)}{2M}k_1\right)\cos\left(\frac{2\pi(2m+1)}{2M}k_2\right), \quad (5)$$

representing 2D DCT basis functions, and $a_{k_1} = a_{k_2} = \sqrt{1/M}$ for $k_1 = 0$ and $k_2 = 0$ respectively and $a_{k_1} = a_{k_2} = \sqrt{2/M}$ for $k_1 \neq 0$, i.e. $k_2 \neq 0$ respectively.

Since it can be observed as a digital image, we assume that the disparity matrix is a K-sparse signal in the 2D DCT domain where $K \ll M^2$, that is

$$f(n,m) = \sum_{i=1}^{K} A_i b_{n,m}(k_{1i},k_{2i}), \quad (6)$$

where A_i denotes the amplitude of the i-th signal component. Disparity matrix has non-zero coefficients at positions (k_{1i},k_{2i}) $i = 1,...,K$ in the 2D DCT domain. This means that only K 2D DCT coefficients of the disparity matrix have significant non-zero values, while other values are equal to zero or negligible.

We further assume that only M_A elements of the disparity matrix are available at positions $(n,m) \in \mathbf{M_A}$ (i.e. $(M-M_A)$ values are missing at random positions). If the signal (i.e. the disparity matrix) satisfies that $K \ll M^2$, according to CS theory, missing samples can be exactly reconstructed if certain conditions are met [9].

B. 2D DCT gradient-based reconstruction algorithm

In the gradient reconstruction algorithm all values at missing samples positions are set to zero. In further iterations these values are considered as minimization variables, and they are varied with a small, appropriately chosen step $\pm\Delta$. For every observed missing value position concentrations of the both 2D DCTs are evaluated, in order to determine the gradient direction, which is defined as the difference of the concentration measures. The missing disparity matrix values are then updated simultaneously in a steepest descent manner. A good starting value of the step can be obtained as:

$$\Delta = \max|f(n,m)|, \ (n,m) \in \mathbf{M_A}. \quad (7)$$

Herein, we adapt the original algorithm for the considered case of 2D signals. Before the algorithm starts, the signal consisted of available signal samples and with zeros at missing samples positions is formed, according to :

$$y(m,n) = \begin{cases} 0, & \text{for } (n,m) \in \mathbf{M}\backslash\mathbf{M_A} \\ f(n,m), & \text{for } (n,m) \in \mathbf{M_A}, \end{cases}$$

where $\mathbf{M}$ denotes the full set of signal positions.

For each iteration k, until the desired precision is obtained, the following steps are repeated:

Step 1: For each missing value at the position $(n,m) \in \mathbf{M}\backslash\mathbf{M_A}$, form two auxiliary matrices according to:

$$y_1^{(k)}(n,m) = \begin{cases} y_1^{(k)}(n,m) + \Delta, & \text{for } (n,m) \in \mathbf{M}\backslash\mathbf{M_A} \\ y_1^{(k)}(n,m), & \text{for } (n,m) \in \mathbf{M_A} \end{cases},$$

$$y_2^{(k)}(n,m) = \begin{cases} y_2^{(k)}(n,m) - \Delta, & \text{for } (n,m) \in \mathbf{M}\backslash\mathbf{M_A} \\ y_2^{(k)}(n,m), & \text{for } (n,m) \in \mathbf{M_A} \end{cases},$$

Step 2: Calculate the finite differences of the signal transform concentration measures [24]

$$g(n,m) = \frac{1}{2\Delta}\left[\mathcal{M}^+ - \mathcal{M}^-\right] \tag{8}$$

where

$$\mathcal{M}^+ = \frac{1}{M^2}\sum\nolimits_{k_1}\sum\nolimits_{k_2}\left|C^+(k_1,k_2)\right|$$
$$\mathcal{M}^- = \frac{1}{M^2}\sum\nolimits_{k_2}\sum\nolimits_{k_2}\left|C^-(k_1,k_2)\right|$$

represent concentration measures. Note that $C^+(k_1,k_2)$ and $C^-(k_1,k_2)$ denote calculated 2D DCTs of two previously defined signals $y_1^{(k)}(n,m)$ and $y_2^{(k)}(n,m)$ respectively.

Step 3: Form the gradient matrix $\mathbf{G}^{(k)}$ of the same size as the disparity matrix $f(n,m)$ with elements defined as follows:

$$G^{(k)}(n,m) = \begin{cases} g(n,m), & \text{for } (n,m) \in \mathbf{M}\backslash\mathbf{M_A} \\ 0, & \text{for } (n,m) \in \mathbf{M_A} \end{cases}$$

with $g(n,m)$ calculated in the Step 2.

Step 4: Correct the values of $y(n,m)$ using the gradient matrix $\mathbf{G}^{(k)}$ with the steepest descent approach:

$$y^{(k+1)}(n,m) = y^{(k)}(n,m) - 2\Delta G^{(k)}(n,m).$$

We multiply the gradient matrix with the factor 2Δ to eliminate the dependence on Δ that appears in (8). By decreasing Δ when the algorithm convergence slows down a high level of precision can be achieved. The proper decrease of the step can be achieved when the oscillatory nature of the adjustments is detected [13].

Let us approximate the numerical complexity of the presented algorithm in one iteration only. Without loss of generality, M is assumed to be a power of 2. For the all missing samples, the total number of additions is, having in mind the complexity of the (fast) 2D DCT algorithm: $(M - M_A)\left[2(M\log_2(M) - 3M/2 + 4)^2 + 4\right]$. The total number of multiplications is approximately: $(M - M_A)\left[2(M\log_2(M) - 3M/2 + 4)^2 + 11\right]$.

IV. Simulation Results and Discussion

In order to illustrate the presented theoretical concepts, several disparity matrices are observed.

Example 1: We consider the compressed sensing scenario, where $M - M_A = 400$ randomly positioned disparity matrix values are missing. Moreover, it is important to emphasize that certain disparity matrix elements have infinite values, corresponding to the background at infinity, and thus we considered them also as unavailable, besides the observed 400 unavailable samples. The number of these elements is 253, 225, 421, 204, 148, 232 respectively, These values represent an additional challenge for the reconstruction algorithm, especially since they are grouped. The corresponding disparity matrices with missing elements are shown in Fig. 3, where missing values are denoted with dark blue color. Light blue denotes infinity while

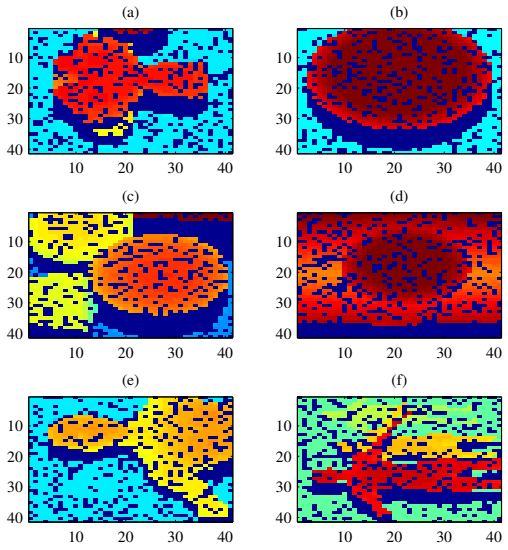

Figure 3. Disparity matrices with $M - M_A = 400$ missing values at random positions

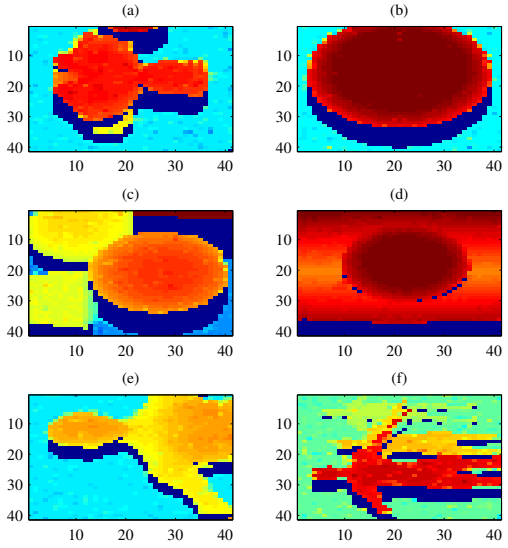

Figure 4. Reconstructed disparity matrices

yellow to red shades denote from furthers to closest object or its parts. We apply the gradient reconstruction algorithm and the obtained results are shown in Fig. 4 (a)-(f). Please note that scanned artificial objects were: a) Blender monkey standard object, b) sphere, c) multiple objects (spheres and cubes), d) small sphere with a background, e) single human, and f) three humans.

Example 2: In order to additionally validate the results, we calculate MSE between the original and undersampled disparity matrices, as well as the MSE between the original and reconstructed disparity matrices. We consider scenarios from 10% to 80% of missing disparity matrix values, with the step of 10%. MSEs are obtained based on averaging the

squared reconstruction errors calculated for 100 independent random realizations of missing values positions in disparity matrices, for each considered percent of missing values. The results are shown in Fig. 5 proving the significant MSE improvement after the reconstruction is performed.

With 80% of disparity missing, sphere object has the MSE of 39.9 dB, while three humans object has MSE of 44.9 dB. CS algorithm demonstrated significant improvement, lowering MSE for both object to 15.7 dB and 15.8 dB respectively, which is significantly better than undersampling of original matrices by 10%. From Fig. 5 it can be seen that difference between undersampled and reconstructed MSE value approximately is the same across whole missing values range. Also, as expected MSE value increases as the number of missing values increases.

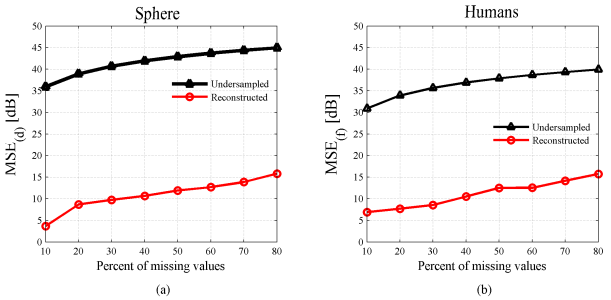

Figure 5. MSE analysis

V. Conclusion

The objective of the paper was to improve the quality of 3D scanner data, by introducing the compressed sensing based signal reconstruction. This in turn enables a more reliable behavior of different smart systems, which can potentially use the 3D scanner's output. In real-life scenarios, the projected patterns are not completely detected by the camera, and thus the maximum number of surface points cannot be achieved. In order to provide a better reconstruction of the scanned surface, the complex 3D reconstruction problem is partly reduced to a 2D problem, and with the use of the compressed sensing algorithms, missing data are filled, obtaining the almost maximum theoretical number of points.

In simulated environment, the proposed method performs very well. The presented results show that even when 80% percent of disparity matrix is lost, the compressed sensing reconstruction algorithm provided a high quality of the reconstructed surface and successfully filled the missing parts of the scanned objects. Despite the time-consumption and the complexity of the CS algorithm, the small resolution of disparity matrix allows future implementation of the algorithm in a mobile robots scanner system. In the future research, the presented CS algorithm would be included in a more complex structured-light scanners with dynamically adaptable density of the projected points, as well as implemented in a parallel processing fashion for the increased computational efficiency.

Acknowledgment

This research is supported by the Croatian-Montenegrian bilateral project "Compressed sensing and super-resolution in surveillance systems based on optical sensors and UAVs".

References

[1] R. Hartley, A. Zisserman, *Multiple View Geometry in Computer Vision, Second edition*, Cambridge University Press, 2004.

[2] G. Shapiro, G.C. Stockman, *Computer vision*, Prentice-Hall, 2001.

[3] E. P. Fotiadis, M. Garzn, A. Barrientos "Human Detection from a Mobile Robot Using Fusion of Laser and Vision Information."*Sensors* , Vol 13(9), pp 11603-11635, 2013.

[4] Y. Cang , J. Borenstein, (2002, May). "Characterization of a 2-D Laser Scanner for Mobile Robot Obstacle Negotiation "*International Conference on Robotics and Automation*, IEEE. 2002.

[5] I. Stancic, J. Music, M. Cecic, "A Novel Low-Cost Adaptive Scanner Concept for Mobile Robots "*Ingeniera e Investigacin*, Vol 34, pp 37-43, 2014.

[6] C. Rocchini, P. Cignoni, C. Montani, P. Pingi, R. Scopigno "A low cost 3D scanner based on structured light "*Computer Graphics Forum*, Vol 20(3), pp 299-308, 2001.

[7] D. S. O. Correra, D. F. Scotti, M. G. Prado, D. F. Sales, "Mobile Robots Navigation in Indoor Environments Using Kinect Sensor. "*Brazilian Conference on Critical Embedded Systems*, IEEE, 2012.

[8] Q. Zhang, S. Li, W. Guo, P. Wang, J. Huang "Refinement of Kinect Sensors Depth Maps Based on GMM and CS Theory," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, Vol 8(5), pp 87-92, 2015.

[9] E. Candès, J. Romberg, T. Tao: "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information,"*IEEE Transactions on Information Theory*, Vol 52(2), pp 489-509, 2006.

[10] E. J. Candès, M. B. Wakin, "An Introduction To Compressive Sampling", *IEEE Signal Processing Magazine*, Vol 25(2), pp 21-30, 2008.

[11] D. Donoho: "Compressed sensing," *IEEE Trans. on Information Theory*, 2006, Vol 52(4), pp 1289-1306, 2006.

[12] R. Baraniuk, "Compressive sensing," *IEEE Signal Processing Magazine*, Vol 24(4), pp 118-121, 2007.

[13] LJ. Stanković, M. Daković, and S. Vujović, "Adaptive Variable Step Algorithm for Missing Samples Recovery in Sparse Signals," *IET Signal Processing*, Vol 8(3), pp 246-256, 2014.

[14] M. Elad, *Sparse and Redudant Representations: From Theory to Applications in Signal and Image Processing*, Springer, 2010.

[15] H. Rauhut, "Stability Results for Random Sampling of Sparse Trigonometric Polynomials," *IEEE Transactions on Information theory*, vol 54(12), pp 5661-5670, 2008.

[16] C. Studer, P. Kuppinger, G. Pope, H. Bolcskei, "Recovery of sparsely corrupted signals," *IEEE Transactions on Information Theory*, Vol 58(5), pp 3115-3130, 2012.

[17] M. Davenport, M. Duarte, Y. Eldar, G. Kutyniok, "Introduction to compressed sensing "*Chapter in Compressed Sensing: Theory and Applications*, Cambridge University Press, 2012.

[18] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, Vol 20(3), pp 301321, 2009.

[19] M. A. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems, " *IEEE Journal of Selected Topics in Signal Processing*, Vol 1(4), pp 586597, 2007.

[20] M. Daković, LJ. Stanković, and I. Orović, "Adaptive Gradient Based Algorithm for Complex Sparse Signal Reconstruction, "*22nd Telecommunications Forum TELFOR*, Belgrade, 2014.

[21] S. Stanković, I. Orović, and E. Sejdić, *Multimedia signals and systems*, Springer-Verlag, 2012.

[22] I. Stanković, I. Orović, and S. Stanković, "Image Reconstruction from a Reduced Set of Pixels using a Simplified Gradient Algorithm,"22nd Telecommunications Forum TELFOR, 2014.

[23] E. Sejdić, A. Cam, L.F. Chaparro, C.M. Steele and T. Chau, "Compressive sampling of swallowing accelerometry signals using TF dictionaries based on modulated discrete prolate spheroidal sequences," *EURASIP Journal on Advances in Sig. Proc.*, 2012:101.

[24] LJ. Stanković, "A measure of some time–frequency distributions concentration," *Signal Processing*, Vol 81, pp 621-631, 2001.