Computer Vision Based Method for Real Time Material and Structure Parameters Estimation Using Digital Image Correlation, Particle Filtering and Finite Element Method

Marcin Tekieli, Marek Słoński

Institute for Computational Civil Engineering
Faculty of Civil Engineering
Cracow University of Technology
{mtekieli,mslonski}@l5.pk.edu.pl

http://www.l5.pk.edu.pl

Abstract. This paper presents the design and implementation of a novel method for real time material and structure parameters estimation. Digital image correlation (DIC) and particle filtering (PF) are used for obtaining the full-field deformations of a structure or model. In order to take into account all advantages of both methods, new marker design is proposed. Particle filtering method is also used in combination with finite element method (FEM) for estimating material and structure parameters, such as Young's modulus, by solving inverse problems. Main algorithm and all of the above methods are implemented in C++. Experiments are carried out on the model of an aluminum frame, using high resolution industrial camera.

Key words: computer vision, digital image correlation, finite element method, particle filtering, sequential Monte Carlo method.

1 Introduction and motivation

The problem of structural material and structure elements identification is an important and difficult class of inverse problems in structural mechanics. Identification is based on a set of measurements on static and dynamic structural responses. Data can be collected by conventional sensors - strain gauges or accelerometers but it requires extensive knowledge of how these sensors work and how they should be mounted on the structure. Connection between each sensor and the data acquisition unit is also required. In this case, data are collected using computer vision methods with high resolution industrial camera, thus the organization of the test stand is much simpler and faster. Optical techniques offer the potential to acquire structure performance data without the need for installation of conventional sensors, lasers or other devices.

The essence of the research is to combine three powerful computational methods - digital image correlation (DIC) for full-field displacement measurements with high accuracy, finite element method (FEM) for structural analysis and particle filtering (PF) for dynamic state estimation and digital image processing.

Digital image correlation is an optical method that is widely used in many areas of science and civil engineering to measure deformation on an object surface [1] and it is relatively easy to use in micro scale for mechanical testing of materials such as steel [2], concrete [3], biomaterials [4] or even paper foils [5].

DIC usage for the displacement measurements in the whole structure or a complex model is rather rare. This is due to difficulties in specifying areas of the specimens because each video frame contains mostly background instead of construction elements. What is more, to ensure the high level effectiveness of this method, the samples would have to have the right texture, which is problematic for some materials used in construction. The proposed method solves this problem by using special markers, very easy to prepare, cheap to manufacture and precede the DIC phase with another step using particle filters to locate markers and use areas that are applicable for monitoring displacement with DIC.

A particle filter, also known as a sequential Monte Carlo method (SMC), is a sophisticated model estimation technique based on simulation. In this method, each distribution is expressed by many of its realizations, and the trajectory of each particle in successive prediction stages is simulated by using the assumed model. At the filtering stage, the resampling with a weight proportional to the likelihood is performed to get a set of particles that represents the filter distribution [6, 7]. In this paper, the same particle filtering is used twice but with different types of particles.

Another method used in this research is finite element method (FEM) - a numerical method for solving differential or integral equations. In this method a model or a structure is divided into an equivalent system of many smaller units (finite elements) interconnected to other elements at points called nodes [8].

Finite element method based Monte Carlo filters for structural system identification [9] ...

2 System Design

The central element of the system is the algorithm responsible for displacement measurements and material or structure parameters estimation. Each video frame taken from camera using uEye Software Development Kit [10] is converted to *IplImage* format from OpenCV Library [11], which was tested during the implementation of the previous project of the computer vision based system for real time traffic sign detection and recognition[12]. Video frames are subjected to preliminary processing in order to prepare it for the next steps of the main algorithm. The application was developed in order to allow its use for different structures (frames, trusses and beams) with any number of measuring points represented by designed markers. Application is fully object-oriented and it was developed in Microsoft Visual Studio 2010 environment. By introducing a number of improvements in the algorithm, it is also optimized to operate in real-time.

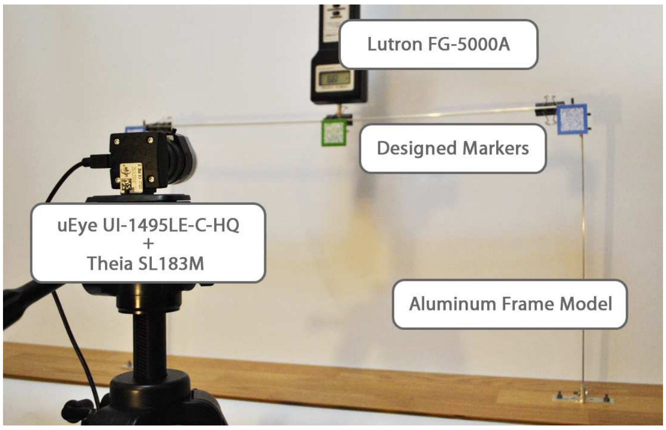

Fig. 1: Test stand with frame model, uEye camera and strain gauge.

Video frames are collected by industrial high-resolution USB camera uEye UI-1495LE-C-HQ equipped with Theia SL183M Ultra Wide Lens. Due to the limitations of the USB interface and very high resolution of each image (3840x2748px), the number of frames processed per second is limited to three. Calculations are performed on a computer with mobile Intel i7 1.86GHz processor and 4GB of RAM.

2.1 Novel Marker Project

Simultaneous use of two different methods - PF and DIC for detection and tracking markers, led to the development of a new marker design, which fully utilizes the advantages of both methods. The marker consists of two elements - single colored frame which allows for rapid searching for a marker using PF and narrows the region of interest only to the area of the marker. Interior of the marker is composed of two kinds of pixels - whites and in the color of the frame. They are set randomly and allow subsequent use of DIC to track the location of the marker and determine its movements. The use of PF allows very fast marker finding, but the tracking of the marker with this method would have relatively small accuracy. The interior of the marker allows to use digital image correlation method, which accuracy is much better. Generation of the described markers is very easy and fast. We can modify the color, size of the markers and change width of the frame. Markers can be adapted to the specific case - the size of the structure and the distance of the object from the camera. The color of the marker may define its features - whether it is a marker in the node of the structure, or a marker placed inside the element. Tests were carried out with markers of size from 10x10mm to 50x50mm and the frame width from 3 to 10mm. Several patterns of the markers are shown in Figure 2.

Fig. 2: Different types of proposed marker model.

2.2 Full-Field Displacement Measurement

The first phase of the algorithm is the detection of the markers. Thanks to the preparation of an application in a fully object-oriented way, work with a different number of markers is comfortable, and each marker is a separate object, which has assigned a set of particles, the position of the marker, the maximum value of the correlation during the displacements measurement and other useful parameters.

Detection of markers is based on a particle filter with particles which are represented by pixels with three main components - the saturation of each color of RGB color space. Reference particle is the pixel with a color similar to the color of the marker e.g. for red markers reference particle can be $p = RGB(255, 0, 0)$.

For each marker a collection of particles covering the entire image is generated with a uniform distribution. To avoid repeated recognition of the same marker, no marker detection is carried out in parallelly, and the occurrance area of next marker's particles is disjointed from the whole area containing marker's particles previously detected. The process of finding the next marker can start only after finding all previous ones.

For image resolution 3840x2748px satisfactory results are obtained already for a population containing only 3000 particles. Searching for markers is a single process and it does not need to be repeated during the algorithm execution. At the test stage, the population of particles for each marker was established at 5000, to ensure the correct detection of each marker.

To calculate the weight of particles, parameters of pixel are used, which are the colors of the three components of the RGB color space. Weight of the particle is higher, if the color of a pixel represented by this particle is more similar to the color of the reference particle. At this stage, the density function for a Gaussian distribution was used, given by following formula:

$$\phi_{\mu,\sigma}(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)} \tag{1}$$

Particles resampling was based on a modified method of the roulette wheel. Weight of each particle is calculated at this stage and then all particle weights are normalized to the $[0.0; 1.0]$ interval. After the weights normalization a new population of particles is created. The value w_t is drawn from the $[0.0; 1.0]$ interval using uniform distribution. From a set of particles, also with uniform distribution, one particle is drawn. The weight of a drawn particle is compared with w_t value and if the weight of the particle is larger, particle is moved to a new population of particles. Otherwise, another particle is drawn and compared with w_t. The process takes as long as the new population of particles is just as large as the population of particles subjected to resampling phase.

In addition, to increase the efficiency of the algorithm, and to avoid convergence to a local minimum, the coordinates of particles are slightly modified with the integer value of the interval $[0.0; resRange]$. The above steps are iterated until all particles assigned to the marker will be moved in the area of selected dimensions (condition of the population concentration).

After the detection phase, measurement of displacements for each of the found markers takes place. At this stage, the image correlation method is used. After the detection phase, each marker is represented by the area containing particles assigned to the marker. Marker's area is narrowed to its interior containing two-color pattern in the middle part of the marker which is suitable for digital image correlation method. Narrowed area is now a reference image, which displacements are monitored. To calculate the correlation coefficient between the model f and a sample g of sizes MxN, the method of Zero Mean Normalized Cross Correlation is used. It is given by the following formula:

$$CC^{ZMN} = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}((f(i,j) - \mu_f) \times (g(i,j) - \mu_g))}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}(f(i,j) - \mu_f)^2} \times \sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}(g(i,j) - \mu_g)^2}} \quad (2)$$

where μ_f and μ_g denote the average luminance of the pattern and the sample. It is the most advanced method of calculation the correlation coefficient, and also the most accurate. Calculation of average luminance for the pattern and the sample improves the results when measurements are made in variable lighting conditions. Best match sample is determined by the maximum value of the correlation coefficient (CC^{ZMN}).

To optimize the algorithm performance, modifications of standard method of correlation are introduced. Search area is not constant and it moves along with the marker. Thanks to that, it is possible to significantly narrow down the search area width and height. It drastically reduces the number of iterations performed during each searching step. Without the introduction of this modification, the value of parameter searchRange, which defines the size of search area, had to be established at 30px. In case of using marker tracking, the value may be reduced up to 5px. In tests the values of 10px and 15px were assumed. The value of this parameter is dependent on the dynamics of changes in the structure and should be inversely proportional to the number of frames processed per second.

2.3 Parameters Estimation

In the second stage, finite element method and particle filtering are used simultaneously to determine structure parameters - in this case Young's modulus of used material. Single particle is represented by finite element method model with particular Young's modulus of the material being used.

Particle filter applied at this stage of the algorithm is very similar to the particle filter used in the markers search phase. The properties of a single particle are represented by a single parameter - Young's modulus, in contrast to a particle represented by the pixel where three values - saturation color component of the RGB color space, are assigned to a single particle.

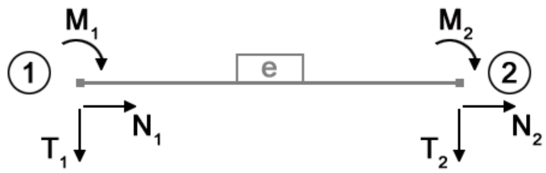

Fig. 3: Frame finite element model.

Each particle is a separate structural model, which is solved by the finite element method using frame elements. Schematic model of this type of element is shown in Figure 3.

$$K = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & \frac{-EA}{L} & 0 & 0 \\ 0 & 12\frac{EI}{L^3} & 6\frac{EI}{L^2} & 0 & -12\frac{EI}{L^3} & 6\frac{EI}{L^2} \\ 0 & 6\frac{EI}{L^2} & 4\frac{EI}{L} & 0 & -6\frac{EI}{L^2} & 2\frac{EI}{L} \\ \frac{-EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -12\frac{EI}{L^3} & -6\frac{EI}{L^2} & 0 & 12\frac{EI}{L^3} & -6\frac{EI}{L^2} \\ 0 & 6\frac{EI}{L^2} & 2\frac{EI}{L} & 0 & -6\frac{EI}{L^2} & 4\frac{EI}{L} \end{bmatrix}, \begin{matrix} A = bh \\ I = \frac{bh^3}{12} \end{matrix} \quad (3)$$

Stiffness matrix K of this element is given by formula (3), where L is length of the element, E is Young's modulus, A is cross-section area and I is element cross section property. FEM algorithm was programmed using Armadillo C++ linear algebra library [13]. Determination of displacement in the structure via FEM boils down to calculating the nodal displacement vector Q from the giving equation:

$$K_g Q = R \quad \rightarrow \quad Q = K_g^{-1} R \quad (4)$$

where K_g is the assembled stiffness matrix of all elements in global coordinate system, and R is a defined force vector.

Because the expected value (correct value of Young's modulus) is unknown, the filter operating time (the number of generations of a set of particles) is dependent on a modified version of the mean square error. Iterations are performed until all values assigned to the N particles (p_{val}) are not in range based on mean value M_{val} and convergence factor C_{val}:

$$[M_{val} - C_{val}, M_{val} + C_{val}], \quad M_{val} = \frac{1}{N}\sum_{i=1}^{N} p_{val} \tag{5}$$

After reaching established convergence, the return value of Young's modulus is calculated as the average value for all particles. In order to confirm the correctness of the results, the tested model of frame was also implemented in the environment for FEM calculations - Abaqus CAE 6.12 [14].

The tests performed with models containing different number of finite elements have shown that in the case of loading the frame model with the concentrated force applied in the middle of the bolt, it is enough to discretize the bolt with only two finite elements (the same discretization can be applied for frame pillars). The number of elements should be increased in the case where the applied force is located at a different point. This is because the point of force application in FEM must be located on the boundary of finite elements.

3 Experimental Results

Preliminary tests are performed on the model of the aluminum frame with markers placed in characteristic points of the structure. Frame model is stressed with a known load at known position, what causes displacements at each node and inside the structure elements. Strength value was determined using digital force gauge Lutron FG-5000A which measures tension and compression in $0.05oz$ or $1g$ resolution with 0.2% accuracy.

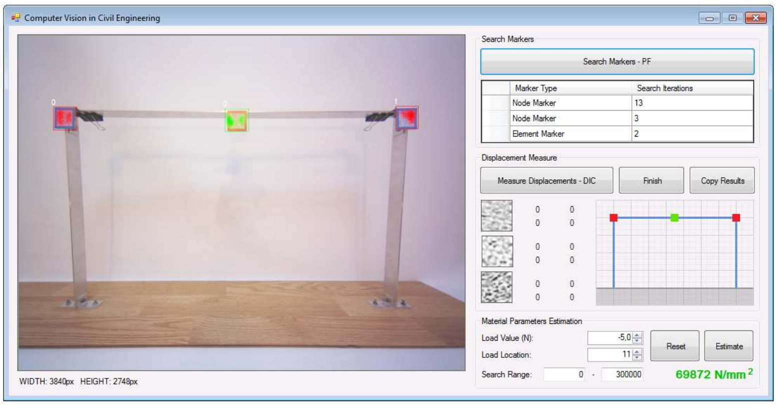

Fig. 4: Application interface.

The first set of tests consisted of measuring the displacement at the point where force of the known values was applied. Force value varied from 1 to $10N$. The results were compared with results obtained from computer simulation of the frame model in Abaqus CAE environment. Resolution of the performed measurements f was set to $0.25mm$. It was determined by measuring the width of the marker on the single video frame. Figure 5 shows obtained results.

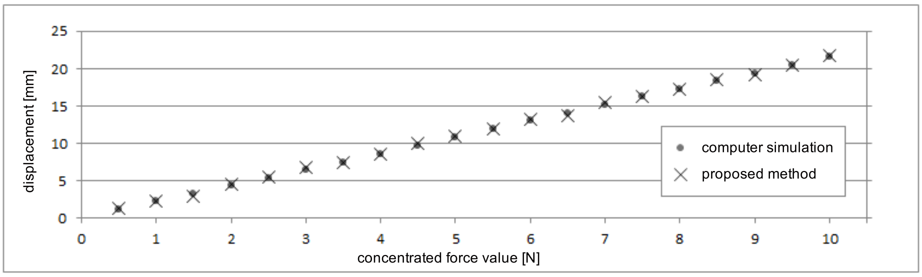

Fig. 5: Vertical displacement in the middle of the frame bolt.

The second set of tests was carried out to estimate the value of Young's modulus of the used material during the structure loading in different points with known concentrated force values. Young's modulus for aluminum is $E_{alum} = 69000N/mm^2$. Parameter estimation was tested in $[0, 300000]$ range. The obtained results are presented in Figure 6.

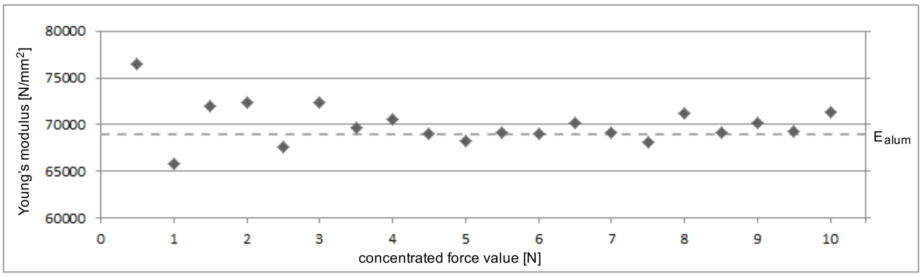

Fig. 6: Young's modulus estimation.

For small values of concentrated force $(1-2N)$ error value is about 6%. With the increasing value of applied force error amounts to 3%.

4 Conclusion and Future Work

The results clearly confirm the effectiveness of the developed algorithm combining two methods that require completely different parameters of marker placed on the structure. We managed to design markers that can be used in case of simultaneous use of both methods and allow us to take measurements at the maximum possible accuracy for used camera.

Currently, work and tests have already been underway to increase the number of parameters estimated at the same time - one of them will be the Poisson ratio. Moreover, the possibilities of location detection of the applied force and the effectiveness of the system with measurement noise added are tested.

Work is also underway on communication module implementation in the RS-232 standard to link our application with the usage of digital gauge meter. This will allow for the estimation of parameters for each video frame, without entering the value of the force manually.

Further testing will be conducted on the actual structure of the bridge over the suburb railway line in Cracow.

References

1. Brémand, F., Malesa, M., Szczepanek, D., Kujawińska, M., Świercz, A., Kołakowski, P.: Monitoring of civil engineering structures using digital image correlation technique. In: EPJ Web of Conferences. Volume 6., EDP Sciences (2010)
2. Savic, V., Hector Jr, L., Fekete, J.: Digital image correlation study of plastic deformation and fracture in fully martensitic steels. Experimental Mechanics **50** (2010) 99–110
3. Skarżyński, Ł., Syroka, E., Tejchman, J.: Measurements and calculations of the width of the fracture process zones on the surface of notched concrete beams. Strain **47** (2011) e319–e332
4. Milosevic, M., Mitrovic, N., Sedmak, A.: Digital image correlation analysis of biomaterials. In: Intelligent Engineering Systems (INES), 2011 15th IEEE International Conference on, IEEE (2011) 421–425
5. Garbowski, T., Maier, G., Novati, G.: On calibration of orthotropic elastic-plastic constitutive models for paper foils by biaxial tests and inverse analyses. Structural and Multidisciplinary Optimization (2012) 1–18
6. Gordon, N., Salmond, D., Smith, A.: Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In: Radar and Signal Processing, IEE Proceedings F. Volume 140., IET (1993) 107–113
7. Kitagawa, G.: Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. Journal of Computational and Graphical Statistics **5** (1996) 1–25
8. Zienkiewicz, O., Taylor, R., Zhu, J.: The Finite Element Method: Its Basis and Fundamentals. Volume 1. Butterworth-Heinemann (2005)
9. Nasrellah, H., Manohar, C.: Finite element method based monte carlo filters for structural system identification. Probabilistic Engineering Mechanics **26** (2011) 294–307
10. IDS-Imaging: uEye Camera Manual (2012) (http://www.ueyesetup.com).
11. WillowGarage: The OpenCV Library (2012) (http://opencv.willowgarage.com/wiki/).
12. Tekieli, M., Słoński, M.: DriastSystem: a computer vision based device for real time traffic sign detection and recognition. In: Artificial Intelligence and Soft Computing, Springer (2012) 608–616
13. Sanderson, C.: Armadillo C++ linear algebra library (2012) (http://arma.sourceforge.net).
14. DassaultSystémesSimuliaCorp.: ABAQUS/Standard User's Manual (2012)