Abstract: - The article is dealing with an approach of the real-time video sequences preprocessing for the global motion removing. In some cases, such a movement is unwanted (e.g. the probe movement during medical scanning) and complicates its further processing. In principle, the presented approach is based on optical flow method. Two methods for the vector field analysis are presented and their result – the global motion vector – is used for motion compensation. Introduced methods are compared even with some simpler and robust methods.

Key-Words: - Image registration, Global motion compensation, Vector field, Optical flow, Image processing, Video sequence processing

1 Introduction

Global motion compensation in a video sequence is a special case of registration of rigid image. Accurate image registration is one of the fundamental problems in digital image processing. We can come across image registration techniques in many scientific disciplines, but this paper is focused on its utilization in the field of biomedical image processing. In the paper, novel methods for real-time motion analysis are presented. They are capable of analyzing a motion which arises from medical scanning. Moreover, the utilization of such methods is possible in each sequence where the significant global motion is present (unwanted translation of camera during video capturing or a scene moving relative to camera without any significant change of the distance from the sensor). The motion can be subsequently eliminated by utilizing a simple image transformation computed from the translational motion. After motion compensation, the video should not contain any undesirable translational motion, which increases its further utilization.

2 Problem Formulation

As mentioned above, the motion compensation in a sequence of consecutive images (frames) in a video fits into the image registration problem. In this case, however, the registration is assumed in its simplest form, when only translation is considered. The situation is complicated by the constraints of real-time processing and the poor quality of input images in sequences.

Currently, there are many methods for image registration [1]. According to the fundamental principle, the methods can be classified into one of the following groups: intensity-based (also called voxel-based) and feature-based methods.

The first category includes methods that are based on the iterative searching for a global extreme in the similarity function $S$ [2] between images $I$ and $J$. The global extreme may be either a minimum or a maximum, depending on the definition of similarity function. For the case of minimum, it can be written as an equation (1) represents image transformation)

$$T_{opt} = \min_{T \in T} S(I, T(J)).$$

The most frequently used similarity functions are SSD (sum of square differences), SAD (sum of absolute differences), correlation techniques CC (correlation coefficient), NCC (normalized correlation coefficient) or mutual information MI (for details see [2], [3]).

Methods from the second group use landmarks in images. On the basis of correspondences between them, the mutual transformation is computed. The transformation can be determined globally for the whole image or for different parts separately. Image registration can be rigid or non-rigid [1]. It depends on the number of degrees of freedom in the transformation matrix.

As an example of feature-based methods, the SIFT [4] (scale invariant feature detection) method can be mentioned. This method performs the robust assignment of correspondences between two images and then the transformation between images can be computed. The method is not sensitive to geometrical distortion, presence of noise and lighting changes. Nevertheless, due to its high complexity this method is computationally quite intensive and its high robustness is not necessary for our purposes (when considering only the translational rigid transformation).

Many systems for motion compensation [5] are also based on optical flow and some of them use similar block diagram as we used (in Fig. 1). Some approaches
change the block diagram by adding some blocks, especially into the phase of global motion post-processing, such as smoothing of global motion vectors with low-pass filter, or utilizing the Kalman filter [6]. At first glance, this post-processing can seem appropriate, but in this specific case, where the motion in input frames is unpredictable, Kalman filtering is inappropriate. Most of similarly oriented papers do not describe in detail how the global motion vector is computed from the local motion vector field, so our paper is focused on this procedure. The proposed two novel methods for global motion vector computation are the main contribution of this paper.

Many papers deal with the utilization of the motion vector field computed during MPEG video coding [7]. This solution is interesting but exceeds the scope of this paper, since it is not directly applicable to biomedical image sequences.

### 3 Problem Solution

The aim of our research is to develop an automatic system that would be able to compensate unwanted motion that is present in so-called free-hand scanning.

Undesirable translational motion occurs between two subsequent images (frames) in a video sequence. This motion must be analyzed and subsequently suppressed, as can be seen in Fig. 2. In Fig. 2 (a) we can see two images from a video sequence before registration. We can see the movement of the stabilized object, caused by global translational motion during the scanning phase. After motion compensation (Fig. 2 (b)), we can see the minimal movement of stabilized object, due to simple image transformation.

Motion compensation is based on the block diagram (Fig. 1), in which we can see the following blocks: feature points extraction, feature points motion estimation, image transformation, and analysis of local motion vectors. The main blocks will be described.

**Feature points extraction:** in periodically repeated time intervals, new feature points are found. The feature points are stored in the memory, and their accurate positions are refined according to the motion in the current frame.

For feature points extraction, the appropriate OpenCV function is used [8]. This function guarantees the desired quality of such points (the minimal value of their own numbers at the given point) and also guarantees the minimum mutual distance (between each pair of points).

**Feature points motion estimation:** The optical flow based on the Lucas-Kanade method [9] was used.

\[
E_x u + E_y v + E_t = 0,
\]

where \(E_x = \frac{\partial E}{\partial x}, E_y = \frac{\partial E}{\partial y}\) and \(E_t = \frac{\partial E}{\partial t}\).

A certain rounded neighborhood of the point is considered, so it gives \(n\) equations with \(n\) partial derivatives \(E_{x1}, E_{x2}, \ldots, E_{xn}, E_{y1}, \ldots, E_{yn}, E_{t1}, \ldots, E_{tn}\) and two variables \(u, v\) which represent the local motion vector of investigated point \((x, y)\) in time \(t\).

**Image transformation:** the image translation is accomplished on the basis of global motion vector.

**Analysis of local motion vectors:** local motion analysis is performed in many places of the processed frame. Due to this, many local motion vectors are obtained. These vectors represent the movement of particular points between two subsequent frames. The local motion vectors generate a vector field, which is characterized by the following properties

- a majority of all vectors have purely translational motion, due to movement in the scanning phase,
- a certain set of vectors is affected by the local motion of objects in the scene (the area is highlighted by red and blue in Fig. 3).

All local motion vectors in the generated vector field are the result of superposition of \(N\) motions

\[
\vec{u}_i = \vec{u}_{i1} + \vec{u}_{i2} + \vec{u}_{i3} + \ldots + \vec{u}_{iN}
\]

which are in further motion compensation deemed as unwanted, disturbing motions. In further computation of global motion vector, their influence must be suppressed.

![Fig. 1: Block diagram of described method.](image)

![Fig. 2: (a) Images from nonregistered sequence (input sequence), (b) images from registered sequence. In (b), stabilization due to image.](image)
3.1 Possibilities for vector field analysis

The following possibilities for the analysis of vector field were tested in order to find the global motion vector on the basis of many local motion vectors. The main contribution resides in the fourth and fifth method (see below), which were designed with respect to a priori known properties of the vector field. Other methods are presented primarily as possibilities and for final comparison.

3.1.1 Method based on arithmetic mean

Intuitively, the easiest method is based on the assumption that a partly random motion is superposed on the principal translational motion. The influence of a random motion would be eliminated when all local vectors are averaged. Therefore, this simplest method calculates the global motion vector as the arithmetic mean of all local motions in the vector field, as can be seen in the following equation

$$\bar{\mathbf{u}}_{avg} = \frac{1}{k} \sum_{i=0}^{k-1} \mathbf{u}_i = \frac{1}{k} \sum_{i=0}^{k-1} (\bar{\mathbf{u}}_{i1} + \bar{\mathbf{u}}_{i2} + \cdots + \bar{\mathbf{u}}_{iN}),$$  \(4\)

$$\bar{\mathbf{u}}_{avg} = \bar{\mathbf{u}}_1 + \frac{1}{k} \sum_{i=0}^{k-1} (\bar{\mathbf{u}}_{i2} + \cdots + \bar{\mathbf{u}}_{iN}).$$  \(5\)

Since the vectors $\bar{\mathbf{u}}_{i2}, \bar{\mathbf{u}}_{i3}, \ldots, \bar{\mathbf{u}}_{iN}$ in Eq. (5) are not random variables, it cannot be presumed that the term $\frac{1}{k} \sum_{i=0}^{k-1} (\bar{\mathbf{u}}_{i2} + \cdots + \bar{\mathbf{u}}_{iN})$ in Eq. (5) will be zero. So the computed global translational vector $\bar{\mathbf{u}}_{avg}$ will not be equal to $\bar{\mathbf{u}}_1$, but it will be affected by this nonzero term. Therefore, the main aim was to design such methods for vector field processing which eliminate this term. The elimination is based on distinguishing between the vectors which represent only translational motion and which are also affected by additional motion vectors.

3.1.2 Method based on median computation for separated components of vector

The key idea of the median utilization method comes from [10]. The method is based on the assumption that purely translational vectors (not affected by disturbing motions) are prevailing in the vector field ($\bar{\mathbf{u}}_i = \bar{\mathbf{u}}_{i1}$ because $\bar{\mathbf{u}}_{i2} = \bar{\mathbf{u}}_{i3} = \cdots = \bar{\mathbf{u}}_{iN} = 0$). The vector field is composed of a set of such purely translational vectors and a set of vectors which are superposed on other disturbing motions.

This method finds the median of local motion vectors to determine the direction of the global translational motion and thus suppress the influence of disturbing motions.

It can be assumed that purely translational vectors prevail. In general, when the median is searched for, the values must be sorted first. Naturally, the biased values (due to superposition of disturbing motions – internal motions, artifacts motion) are located at both the beginning and the end of the sorted list. These biased values are ignored and they do not affect the calculation of global motion vector $\bar{\mathbf{u}}_{med1}$

$$\bar{\mathbf{u}}_{med1} = (\text{med}(\{u_{i1}\}), \text{med}(\{v_{i1}\})).$$  \(6\)

3.1.3 VDF method [11]

If the median is calculated for each component separately (see the previous section), it does not exactly represent the correct definition of median. The result of correct median calculation has to be equal to one of the input values. According to Eq. (6), a new vector can be constructed. In this case it is not a big problem, but in some applications this is undesirable and so the median computation in the vector field is investigated [11]. In this project, the VDF [11](Vector Direction Filters) method was used for comparison with other methods. The VDF method selects as the median $\bar{\mathbf{u}}_{med2}$ such a vector for which the sum of angles contained between the $i$-th vector and all the other vectors is minimal according to the equation

$$\alpha_i = \sum_{j=0}^{k-1} \cos^{-1} \left( \frac{\bar{\mathbf{u}}_{i} \cdot \bar{\mathbf{u}}_{j}}{|\bar{\mathbf{u}}_{i}| \cdot |\bar{\mathbf{u}}_{j}|} \right), \quad \text{pro } i = 1, 2, \ldots, k - 1,$$

$$\bar{\mathbf{u}}_{med2} = \bar{\mathbf{u}}_{i}, \text{ where min } (\{\alpha_i\}).$$

There exist certain extensions of this method, but they are beyond the scope of this paper (for example WVDF, SWVDF [11]). The methods mentioned above have been designed for median filtering of color images (3D vectors are used here – RGB, or HSV). In this paper, the VDF method was tested and its results were compared with the results of other methods.

Fig. 3: Vector field based on optical flow. The highlighted parts of vector field show vectors disturbed by local motion in scene.
3.1.4 Method based on vector angle histogram

The first method we propose is based on angle histogram and is designed to consider the information about vector field which is known a priori. The method is based on the assumption that most vectors in the vector field are oriented in parallel with global translational motion (these vectors are not significantly superposed on disturbing motions). The proposed method is therefore based on the histogram of the transformed angles of vectors in the vector field.

The main procedure starts with the computation of angles \( \text{angle}(\bar{u}_i) \) for each vector \( \bar{u}_i \) in the vector field \( \bar{U} = \{\bar{u}_i\} \). For convenience, each computed angle is rounded and mapped (Eq. 9) to a certain integral range; in our case a range between 0 and 36 was used, giving 36 discrete angle values. Such a range was chosen experimentally in order to achieve good resolution in the histogram.

\[
h_i = \lfloor \text{angle}(\bar{u}_i)/10 \rfloor \quad (9)
\]

The histogram is computed in order to determine the most frequent angle. All \( \{h_i\} \) values are included in the histogram. Moreover, the values \( \{h_i + 1\} \) and \( \{h_i - 1\} \) are included with half the weight for the sake of minimizing the transformation error. The maximum value in histogram \( \varphi' = \max((h_i)) \) defines the most common direction in the vector field. In further processing, only vectors \( \bar{u}_i \) which are similar to the most common direction \( \varphi \) are considered (\( \varphi \) is the inverse transformation of \( \varphi' \))

\[
\bar{U}' \subseteq \bar{U}, \bar{U}' = \{\bar{u}_i \mid \text{abs}(\text{angle}(\bar{u}_i) - \varphi) < \text{thresh}_\varphi \} \quad (10)
\]

The global motion vector \( \bar{u}_{m1} \) is then computed as the average mean of the subset \( \bar{U}' \),

\[
\bar{u}_{m1} = \frac{1}{k} \sum_{i=0}^{k-1} u_{i}' \quad (11)
\]

3.1.5 Method based on vector norm

A significant modification of the previous method led to the design of the second method we propose. It is based on the computation of the norm of vectors (vector length). First, the Euclidian norm of each vector in the vector field is computed

\[
||\bar{u}_i|| = \sqrt{u_i^2 + v_i^2} \text{ for } i = 0, 1, ..., k - 1 \quad (12)
\]

According to the characteristics of vector field, most vectors correspond to purely translational motion, and their norms are approximately the same as the translational motion length. So in the set of norms, such value will prevail. Therefore, the median is selected from the set of vector norms. Due to the median utilization, the sample \( l \) which probably corresponds to the length of translational motion will be selected. The advantages of the median have been described above. Only vectors with a norm similar to \( l \) are considered in further processing

\[
\bar{U}' \subseteq \bar{U}, \bar{U}' = \{\bar{u}_i \mid \text{abs}(||\bar{u}_i|| - l) < \text{thresh}_l \} \quad (13)
\]

The resulting global motion vector \( \bar{u}_{m2} \) is then computed as the average vector in a subset \( \bar{U}' \),

\[
\bar{u}_{m2} = \frac{1}{k} \sum_{i=0}^{k-1} u_{i}' \quad (14)
\]

4 Results

In further evaluation, the registration errors will be perceived as errors between manually registered sequence and sequences registered by the presented methods. The registration error is evaluated in every thirty-fifth frame (i.e. approximately every second of the sequence). There is significant translational motion in testing data, the average motion in 1s interval was 28px.

The utilized test set of sequences was not large, because of the difficulties arising from the manual registration (the global translational vector had to be determined for each single image). However, suitable test sequences were chosen to cover the majority of possible cases – from the negligible motion to very fast motion during scanning, from a small amount of disturbing artifacts to heavily disturbed sequences. The accuracy and robustness of each method could then be compared.

In Fig. 4, the histogram of compensation errors is shown. Relative frequencies of errors obtained in the motion compensation process are plotted there. The first bar of the histogram represents the relative frequency of compensation error in a range from 0 to 4px (it is error within 1s of record – minimal error). It can be seen that our first proposed method (referred to as method 4 above) is the most precise, because the minimal registration error mentioned is achieved in more than 60% of all registrations. Other methods achieve this minimal error only in approximately 40% (in other cases, the registration error is greater than 4px). Other bars represent relative frequencies of motion compensation errors in ranges from 4 to 8px, 8 to 12px, etc.

If a compensation error of less than 7px is acceptable (again calculated as error among 35 frames), our first proposed method (method 4) was again the most precise (Fig. 8). By using this method, in 80% cases the registration error was acceptable (lower than the 7px considered), while when using other methods the registration error is acceptable only in 50% of cases.

In Fig. 6 the progress of registration error during a video sequence can be seen. A selected part of the
sequence shows an interval with significant motion during scanning. Registration errors of simple methods (methods 1, 2 and 3) are relatively high (the maximum error reaches 18px). Advanced methods (methods 4 and 5) perform a better registration since the registration error did not exceed 8px in the interval displayed.

All the performed tests show that the proposed methods (methods 4 and 5) provide a more accurate processing of the vector field in comparison with the proposed simple methods. Thus the proposed methods are better in the estimation of the final translational vector in the image.

The proposed methods were also compared with the pyramidal technique implemented in TurboReg – the common available tool for Java [12]. TurboReg registration method is based on minimizing of SSD (Sum of Squared Differences) [2]. This method is suitable for most cases of registration, but for registration of ultrasound image sequences it do not work properly. The precision of the registration is affected by artifacts described above. When the amount of artifacts in video sequence is low, the registration is quite precise, comparable with proposed methods. Fig. 5 shows the comparison of compensation errors for proposed method no. 4 and for TurboReg. It can be seen, that TurboReg precision is slightly lower than best of proposed methods.

In Fig. 7 the comparison of progress of registration error during the video sequence can be seen (the video sequence contain many artifacts). It can be seen, that compensation error, when using TurboReg, is higher and in some moments the error exceeds 10px.

Another important drawback of the TurboReg method lies in computational complexity – the registration of two images takes more than one second. TurboReg is unusable for real-time image processing.

Using the same hardware and the same testing data, the time required for motion compensation by our proposed methods varies from 14 to 17ms, i.e. motion compensation is capable of processing video sequences at a speed of up to 59fps. Moreover the time complexity of proposed algorithm is not dependent on the video sequence resolution. The time complexity depends on the count of local vectors in the vector field, which can be regulated in feature point detection block. It enables to use our method even in high resolution video sequences.

Obviously, the proposed methods are much more accurate in comparison with the other simple ones. The proposed methods achieved a registration quality similar to the pyramidal technique used in TurboReg. However, the performance of the proposed methods is much better because they can process video sequences in approximately 59 fps.

In the future, the results could be improved by preprocessing the motion vector field. On the other hand, this would probably require a trade-off between the accuracy improvement gained and the decrease in the performance of the methods.

5 Conclusion
In this article, two advanced methods for translational motion compensation in video sequences have been presented. Their accuracy and time performance have been tested and compared with other methods.

Acknowledgment
This work was prepared with the support of the MSMT project No. 2B06111.

References: