UDC 004.9

Olena A. **Arsirii** [1], Doctor of Technical Sciences, Professor, Head of the Department of Information Systems, E-mail: e.arsiriy@gmail.com, ORCID: https://orcid.org/0000-0001-8130-9613
Yuliia L. Troianovska [1] Senior Lecturer at Information System Department,
E-mail: troyanovskaja@gmail.com, ORCID: https://orcid.org/0000-0002-6716-9391
Iryna A. Prykhodko [1], Master of Computing Sciences, Department of Information Systems,
E-mail: ir.prikhodko@gmail.com, ORCID: https://orcid.org/0000-0002-7846-9867
Diana Yu. Kotova [1], Bachelor of Computing Sciences, Department of Information Systems,
E-mail: kotowstrider@gmail.com, ORCID: https://orcid.org/0000-0002-1067-0433
Odessa National Polytechnic University, Odessa, Shevchenko Avenue, 1, Ukraine. 65044

ARCHITECTURAL OBJECTS RECOGNITION TECHNIQUE IN AUGMENTED REALITY TECHNOLOGIES BASED ON CREATING A SPECIALIZED MARKERS BASE

Abstract. *The paper proposes a method for recognizing architectural objects when creating augmented reality mobile applications based on building a database of specialized markers. The main method of augmented reality technology for the recognition of architectural objects was chosen - the technology based on special markers. The range of pattern recognition algorithms suitable for the task is highlighted. These are algorithms based on the selection of key points of images and their descriptors. The most important aim is the stable recognition of architectural objects upon mobile applications for augmented reality-type digital guide creation based on specialized markers. The scientific basis of the research is a systematic approach in the analysis of the considered markers recognition algorithms, machine learning for the development of a database of marker images and AO recognition are used. The technique consists of the following steps: processing images of architectural objects with the aim of identifying key points, obtaining descriptions of selected control points as descriptors, creating AR-metadata that correspond to architectural objects, organizing joint storage in the local database of descriptors and their corresponding metadata, visualizing the architectural object and AR metadata. To implement the stages of processing images of architectural objects and obtain descriptors of key points, algorithms for extracting key points on images, such as SIFT, MSER, SURF, RIFF, RF, are analyzed. It is shown that these algorithms are invariant to scaling, rotation, as well as resistant to changes in light, noise and viewing angle. A comprehensive use of them for processing architectural objects with the aim of obtaining descriptors of reference points was proposed. To ensure stable recognition of AO according to the developed methodology based on machine learning for processing architectural objects with the aim of obtaining descriptors of key points, it was proposed to create an additional module using an ordered stack. The launch sequence and the number of algorithms can be changed.*

Keywords: *information technology; intellectual analysis of data; augmented reality; AR-technology; marker methods of recognition*

Introduction and statement of research problems

At the present stage of development of information technologies (IT), a special place is occupied by software tools for the automated creation of augmented reality systems (AR augmented reality) [1]. The components of augmented reality "artificially" change the world around them through the composition of real and synthesized virtual objects. The emergence and distribution of mobile communication devices with a sufficiently large power of processors and video cameras with high resolution allowed creating the AR applications on mobile platforms. One of the promising practical ways of mobile AR applications development is a creation of IT for the recognition of various architectural objects (AO) [2] in order to obtain their description and / or reconstruction in the form of lost fragments or type of AO before restoration. In modern digital virtual guides, the complex data obtained from a GPS, a gyroscope and a compass that are embedded in a mobile phone is used to determine the location of an AO. The activation of AR application occurs hen the coordinate ata f the r's bile phone coincides with the data about the AO in a special database with AR-metadata. However, the modern mobile device's geolocation system, which has an error of up to 5 m, is practically useless for building digital guides, especially in cases where the AO are located close to each other. That's why at the present stage, IT is used to build mobile AR tourist guides, which combine AR technology with image recognition systems [22]. To ensure the operation of the AR-guide, it is necessary to input an image of an AO using a mobile phone camera and, match the resulting image with the standards (markers) stored in a special database available on a mobile device or server after preprocessing. Such a database combines marker images of AO and metadata that

contains a description and / or AR reconstruction for the entered AO. Various technologies are used to build a database of specialized markers.

Analysis of existing scientific achievements and publications

The existing IT for a database of AO markers [3-5] creation and the following groups were highlighted:

1. *The technology based on the construction of key points on the virtual grid* works using special recognition algorithms, where a virtual "grid" is superimposed on the surrounding landscape, captured by a camera. Software algorithms find some key points on this grid that determine the coordinates of the object to which the virtual model is "attached". The advantage of this technology is that the real-world objects work as markers and there is no need to create some special visual identifiers. However, the definition of geocoordinates on a virtual landscape grid requires large memory resources and a mobile device processor is also not reliable enough.

2. *The technology based on special markers or tags,* is convenient because they are easier recognized by an AR application and provide an adaptive binding to the location for the virtual model. Two options are possible for implementation of this technology:

a) *cloud realization*, when the marker base is located on the server, and not on the mobile device, and access is provided via access keys. In this case, the base may contain multiple duplicate markers for better recognition of AO. Disadvantages: Internet connection required, time of the recognition increases;

b) *local realization*, when the markers are stored in a local database on a mobile device, which allows to increase the speed of work and does not require an Internet connection. However, with the help of such a database, it is possible to recognize signs on buildings or QR codes successfully, but not real images of complex AO. The main problem of this approach compared to the cloud implementation is that for the AO markers recognition it will be necessary to use a sufficiently large amount of limited resources of a mobile device.

The publications analysis [6-21] made it possible to highlight the following methods and algorithms that show good results while processing specialized images containing an AO in order to extract key points on them:

– *SURF (Speeded Up Robust Features)* [7-9; 14] is based on key points search and descriptors creation that are invariant to scaling and rotation. At the same time, for each point the gradient of the maximum brightness change and the scaling factor according to the Hessian matrix or Haar wavelets are considered;

– *RIFF (Rotation Invariant Fast Features*) [8; 14] is based on the radial and tangential decomposition of gradient histograms followed by ring processing. The resulting descriptor is also invariant to scaling, rotating and light changing;

– *RF (Random Forest*) [15]. The main idea of RF is to train the recognition system based on the collected statistics of decision distribution by constructing a forest of random trees;

− *SIFT (Scale Invariant Feature Transform)*[16] is based on obtaining points that are invariant to scaling and rotations of the image, resistant to light changes, noise and changes of the observer position;

− *MSER (Maximally Stable External Regions)* [17] is based on extracting the total number of corresponding image elements and contributes to the wide comparison of baselines.

The purpose and objectives of the research

The purpose of the research is to develop a methodology of recognizing architectural objects upon mobile applications of augmented reality-type digital guide creation based on building a database of specialized markers.

To achieve the goal, the following tasks were solved:

1) the method of recognizing AO upon mobile applications augmented reality-type digital guide application creation using machine learning;

2) the capabilities of the basic algorithms and methods for extracting key points on the image were analyzed in order to build descriptors;

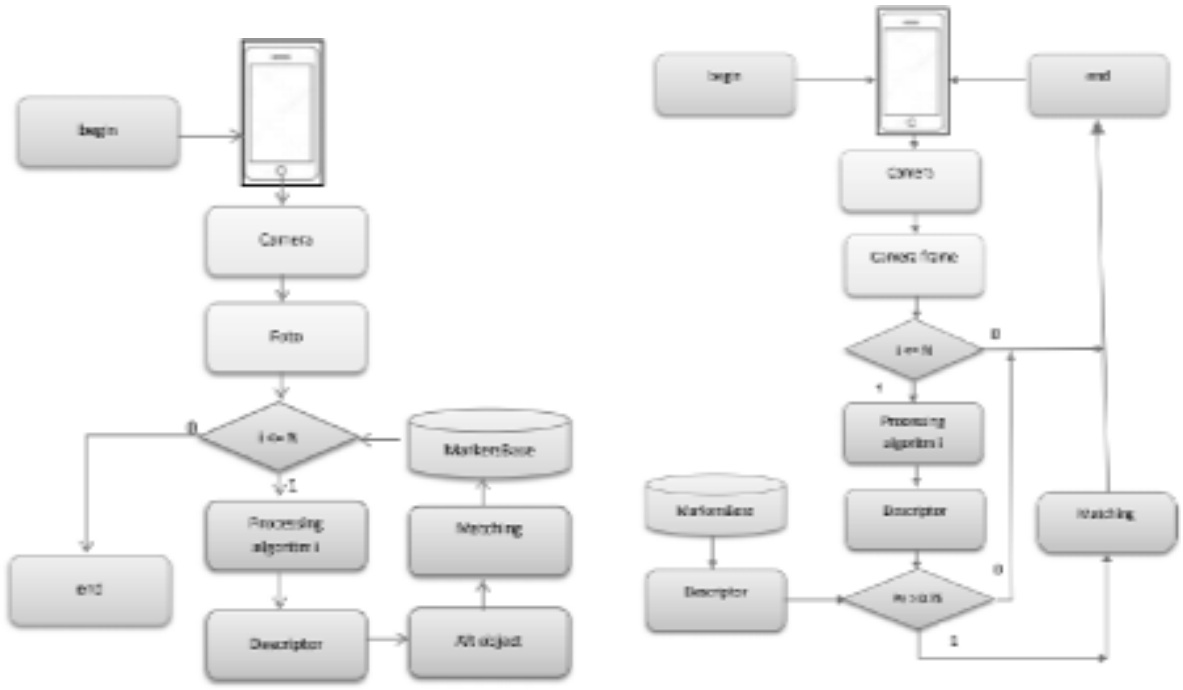
3) the use of a stack of ordered processing algorithms for constructing key points descriptors when creating a database of special marker images of AO was proposed.

Research methods. As a scientific basis of the research, a systematic approach in the analysis of the considered marker (image) recognition algorithms, machine learning for the development of a database of marker images and AO recognition are used.

Presentation of the main research material

The method of recognizing AO upon digital guide application creation.

In this work on creating the method of AO recognition in mobile applications an approach based on machine learning has been used. When implementing this approach, it is necessary to support two modes of mobile application operation: creating a database of specialized AO markers and recognizing an AO image entered by the user in order to demonstrate the corresponding AR from the prepared base (Fig. 1)



a b

Fig. 1. The method of AO recognition in mobile applications with augmented reality:
a – mode of creating a database of specialized markers; b – mode of recognition of user-entered AO image

The mode for creating a database of specialized AO markers generally includes the following steps:

1) receiving the original image using a camera in the mobile phone;

2) pre-processing of the original image $I(xR_1,yR_1)$ with the view of detect the key points of the image $I(xL_1,yL_1)$;

3) compiling descriptions of key points (definition of a descriptor);

4) repeating steps 1 and 2 for all processing algorithms available in the database;

5) receiving metadata by linking descriptors with an AR block that is associated with the original image and additionally visualized when the camera is aimed at a recognized AO;

6) adding metadata to special markers database.

Mode of user-entered AO image recognition generally includes the following steps:

1) receiving the original image and its processing according to steps 1-3 of the base creation mode;

2) calculation of the distance between the received descriptor and descriptors from the specialized markers database. The winner is the descriptor, which has the smallest distance between points in a pair;

3) if the calculated distance is less than some experimentally threshold P_e , then an AR-block is extracted from the metadata corresponding to the winner descriptor, which is visualized additionally with the image of AO;

4) if the calculated distance is greater than some threshold, then the original image is pre-processed by the following algorithm from a stack of algorithms, the repetition of steps 1,2. Further verification, corresponding to step 3, is repeated;

5) the AO recognition process ends if the corresponding AR-block is found or the stack of existing preprocessing algorithms is exhausted.

It should be noted that for the formation of the preprocessing algorithm stack, it is necessary to analyze existing algorithm key with regard to the requirements for building mobile digital guides such as: invariance to scaling and rotation, resistance to changes in light, noise and angle. In addition, in analyzing the algorithms their ability to recognize large enough AOs that have complex geometric shapes should be taken into account. The recognition requirements are tightened by the fact that it is necessary not only to determine the AO but to highlight a label on it for the subsequent visualization of metadata with an AR block.

Analysis of the capabilities of the main algorithms for extracting key points.

Taking into account the above information, a comparative analysis of key points selection algorithms can be conducted.

SIFT algorithm (Scale Invariant Feature Transform). Most angle search algorithms do not

depend on the rotation of the object, since even if the image is rotated; the definition of angles remains possible. The angle may stop being an angle if the image is scaled. For example, by approximating a square-shaped element, we obtain an image of a straight line, which is a side of the original figure. The solution to this problem is to use a scale-invariant transformation of signs. The SIFT algorithm consists of five main steps [7].

1) *Scale-spatial detection of extremes.* From theory it is known that it is impossible to use the same window to detect key points with different scales. In this case, a positive result can be obtained in the case of a small angle. To detect large angles, it is necessary to use large size of windows. The problem is solved on the basis of scale-spatial filtering with the calculation of Laplace operator of the normal distribution (Gaussian). Gaussian acts as a scaling option. For example, a Gauss core with a low σ gives a high value for a small angle, while a Gauss core with a high σ is suitable for a larger angle. Thus, you can find local maxima in scale and space, which give us a list of values.

It means that a potential key point exists in *(x, y, σ)* to scale:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y),$$

where:

L – the value of the Gaussian at the point with coordinates (x, y);

σ – blur radius;

G – Gauss core;

I – source image value;

* – convolution operation.

However, the calculation of a Gaussian is resource intensive, so the SIFT algorithm uses a difference of Gaussian (DoG), obtained as a difference in Gaussian image blur with two different σ (for example, σ and kσ).

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) =$$

$$= L(x, y, k\sigma) - L(x, y, \sigma). \quad (1)$$

Scale invariance is achieved by finding key points for the original image, taken at different scales. For this, the Gaussian pyramid is built: the entire scalable space is divided into some sections, the octaves, and the part of the scalable space occupied by the next octave is twice as much as size of the occupied part (Fig. 2)

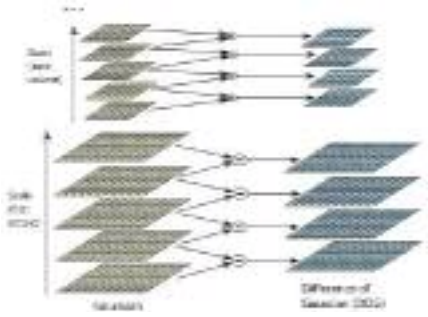


Fig. 2. Pyramid of Gaussians and their differences in the SIFT algorithm [10]

Further, we will consider a point as a key if it is a local extremum of the difference of Gaussians. In each image from the DoG pyramid, extremum points are searched. Each point of the current DoG image is compared with its eight neighborhoods and with nine neighborhoods in the DoG, which are one level higher and lower in the pyramid. If this point is more (less) than all the neighborhood, then it is taken as a point of local extremum. If this is a local extremum, this is a potential key point, which means that the key image is best represented at this scale.

1) *Localization of the key point*. Once potential key points are found, they need to be refined in order to get more accurate results. For example, in SIFT, the expansion of the Taylor scale is used to obtain more precise location of the extremes, and if the intensity at these extremes is less than the threshold value, it is rejected.

The difference of Gaussians has a higher response to the edges of the shapes, so the edges must also be removed. To do this, is used an approach similar to the calculation of the angular Harris detector. SIFT algorithm is used as Hessian matrix (H) with size is 2x2 to calculate the principal curvature. At the same time, for the edges one eigenvalue (by Harris detector) is larger than other, therefore is used as simple comparison function. If the obtained ratio is bigger than the threshold value, this key point is discarded. Thus, at this step, any low-contrast key points and boundary key points are eliminated, while the remaining ones turn out to be the points of greatest interest.

2) *Orientation distribution*. Orientation is now assigned to each datum to achieve a constant image during rotation. A neighborhood is taken around the location of the reference point depending on the scale, and the magnitude and direction of the gradient is calculated in this area. An orientation histogram is created with 36 cells, spanning 360 degrees. (It is measured by the magnitude of the gradient and by the Gaussian round window with σ equal to one and a

half times the key point scale). The highest vertex in the histogram is taken, and any vertex above 80 % is also taken into account for calculating the orientation. Thus, control points are created with the same location and scale, but in different directions. This contributes to the stability of the comparison.

3) *Key point descriptor*. Descriptor is some information about the neighborhood of the reference point. As a rule, a 16x16 neighborhood is chosen around a key point. It is divided into 16 4x4 sub-blocks. For each sub-block, an orientation histogram is created, consisting of 8 cells. Thus, a total number of 128 cell values are available, which are represented as a vector to form a key point descriptor. In addition to this, some measures were taken in the SIFT algorithm [9] to ensure resistance to changes in light, rotation, etc.

4) *Key points comparison*. Key points between two images are matched by determining the nearest neighborhood. But in some cases the second closest match may be very close to the first one. This may be due to noise or for some other reason. In this case as a measure of proximity the ratio of the closest distance to the second closest distance is taken. If it is more than the threshold value of 0,8 then they are rejected. According to studies, the use of such a modification eliminates about 90 % of false matches, with the simultaneous loss of only 5 % of correct matches.

Conclusion: Descriptors resulting from SIFT are local and are based on the appearance of the object at certain points of interest and do not depend on the scale and rotation of the image. They are also resistant to changes in light, noise and minor changes in the point of view (viewing angle). In addition to these properties, they are relatively easy to remove and allow you to properly identify an object with a low probability of inconsistency. SIFT descriptors provide a relatively easy search in a small database of local objects, but, nevertheless, the large dimension of the algorithm can be a problem, and probabilistic algorithms such as k-d-trees are usually used, with the best cell search at the beginning. Recognition can be performed in real time for small local JSC databases using a modern mobile device.

SIFT algorithm modifications. Modification of the SIFT algorithm are in the interest - PCA-SIFT [11; 12], in which the resulting set of SIFT descriptors reduces the dimension of the vectors to 32 elements by principal component analysis (PCA). According to PCA-SIFT the magnitude and orientation of the gradient at the initial stage are similarly calculated. Only for each key point is considered a neighborhood of 41 × 41 pixels with a center at a special point. In fact, the map of gradients in the vertical and horizontal directions is constructed. As a result, it turns out a vector containing 2×39×39=3042 elements, and then the SIFT descriptor is built according to the scheme.

GLOH algorithm (Gradient location-orientation histogram) [13] is also a modification of SIFT algorithm which is built to improve the reliability of key points selection. The SIFT descriptor is calculated, but the polar grid is used to divide the neighborhood into the so-called “bins”: 3 radial blocks with radii of 6, 11, and 15 pixels and 8 sectors. As a result, vector containing 272 components is projected using PCA into a space with a dimension of 128.

SURF algorithm (Speeded Up Robust Features). SURF is based on properties similar to SIFT and was developed by lower computational complexity [8]. In addition, there exists also a vertical version of the descriptor (U-SURF) which is calculated even faster without being an invariant of image rotation and is better suited for applications where the camera remains static in the horizontal position.

The first step of processing using the SURF algorithm is to set up a reproducible orientation based on information from the space around the key point. In the second step is created a square region aligned to the chosen orientation and the SURF descriptor is extracted from it [14]. Some processing steps of SURF in more details are reviewed [23].

1) *Distribution of the orientation*. To be invariant to the rotation the algorithm identifies reproducible orientation for points of interest based on calculating the Haar wavelet responses in the x and y directions in a circular neighborhood of radius $6s$ around the point of interest, where s is the scale within which the point of interest was detected (for the reasons of symmetry and discretization the allowable sizes starting from the minimum: 9; 15; 21; 27 and etc., with a step of 6). However, in practice for large scale step 6 is smaller. Therefore, SURF breaks the whole set of scales into so-called octaves. Each octave covers a certain range of scales. The wavelet responses are calculated at this current s-scale. Accordingly, at large scales the size of wavelets increases. For the invariance of the calculation of the key point descriptors, which will be discussed below, it is necessary to determine the prevailing orientation of the brightness differences at the key point. This concept is close to the concept of a gradient, but SURF uses a slightly different algorithm for finding the orientation vector. At first, the point gradients in pixels adjacent to the key point are calculated. Pixels in a circle of radius 6s around a key point are taken into the consideration. Where s is the scale of the key

ISSN 2663-0176 (Print)
ISSN 2663-7731(Online)

point. For the first octave the points from a neighborhood of radius 12 are taken. To calculate the gradient, Hear filter is used. The filter size is taken equal to 4s, where s is the scale of the key point.

After the wave resolution responses are calculated and weighted with a normal distribution (σ = 2.5 s) the responses are represented as vectors in space with a horizontal response force along the abscissa axis and a vertical response force along the ordinate and are centered on the point of interest. The dominant orientation is estimated by calculating the sum of all responses in a sliding orientation window covering the $\frac{\pi}{3}$ angle. Horizontal and vertical responses by the window are summarized. Then two summarized answers give a new vector. The longest of these vectors gives an orientation to the point of interest. The size of the sliding window is a parameter that was chosen experimentally. Small sizes correspond to a single dominant wavelet response. Large sizes give maxima along the length of the vector which is not expressed. Both lead to unstable orientation of the area around the key point. For the vertical version of the SURF algorithm – U-SURF orientation distribution isn't carried out.

2) *Components of descriptor*. To extract the descriptor, the first step is to construct a square region centered around the anchor point and oriented along the direction (orientation) chosen in the previous section. For the vertical version, this conversion is not necessary.

The size of the potential window is 20s. The region is regularly divided into smaller 4 × 4 square subregions. This allows to save important spatial information. For each subregion, several simple functions are calculated at 5 × 5 regularly located sample points. For simplicity, we call Haar wavelet response in the horizontal direction and dy Haar wavelet response in the vertical direction (filter size 2s). "Horizontal" and "vertical" are defined here depending on the chosen orientation of the point of interest. To increase resistance to geometric deformations and localization errors, the responses of dx and dy are first weighed with a normal distribution (σ = 3,3 s) with the center at the point of interest.

Then, the wavelet responses of d_x and d_y are summed over each sub region and form the first set of records in the feature vector. To obtain information about the polarity of intensity changes, we also extract the sum of the absolute values of the answers – $|d_x|$ and $|d_y|$.

Consequently, each subregion has a four-dimensional descriptor vector for its basic intensity structure:

$$v = (\Sigma d_x, \Sigma d_y, \Sigma|d_x|, \Sigma|d_y|). \quad (2)$$

The result is a descriptor vector for all 4 × 4 subregions of length 64 (SURF). The wavelet responses are invariant to the offset in the light. Invariance to contrast (scaling factor) is achieved by turning the descriptor into the unit vector.

Fig. 3 shows the descriptor properties for three distinctly different intensity models of the image by subregions. It is possible to imagine a combination of such local intensity patterns which leads to a distinctive descriptor.

To reach this SURF-descriptors, it is necessary to experiment with big and small amount of wavelet functions, using d2x and d2y, higher order wavelets, principal component method, median values, average values, etc. Then the number of sample points and subregions experimentally varies. The 4x4 solution for the subregions helped to achieve the best results. Smaller subregions turned out to be less reliable and increased the matching time too much. On the other hand, a short descriptor with 3 × 3 subregions (SURF-36) works worse, but allows very fast matching and is still quite acceptable compared to other algorithms in the literature [10]. The SURF descriptor is a set of 64 or 128 numbers for each key point in the SURF-128. Fig. 4 shows some of these results comparing the accuracy of coincidences of descriptors constructed using the described SIFT algorithm and its modifications and the SURF ruler algorithms.

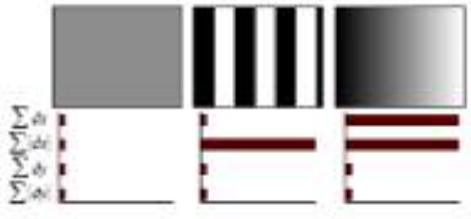
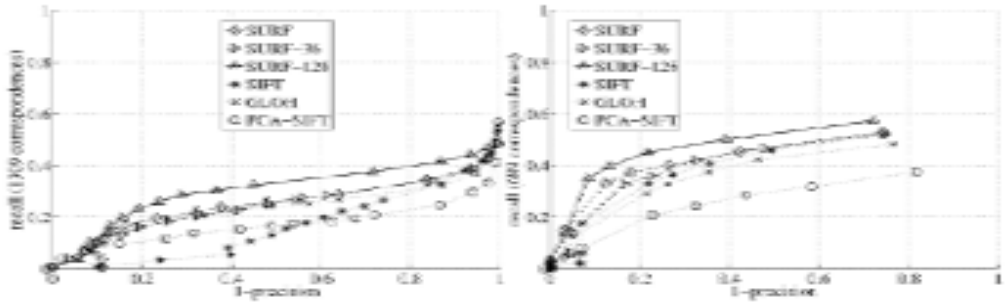


Fig. 3. Descriptor subregion entries displays the character of the base intensity pattern [23]

Conclusion: SURF has many additional features to improve processing speed at each stage. The analysis shows that it is 3 times faster than SIFT. SURF is good at processing images with motion blur and rotation, but it doesn't go well with changing viewpoints and illumination.

RIFF algorithm *(Rotation Invariant Fast Features*). RIFF using the approximate conversion of the radial gradient is used to significantly reduce the calculation time of feature extraction. With this algorithm, descriptors are built, for example, for aerial photographs taken from platforms that are subject to a high degree of rotation due to sudden maneuver, scaling, changes in light and noise.



a b

Fig. 4. A graph of coincidence accuracy for various processing algorithms and two different matching strategies tested with a 30 degree change of view compared to current descriptors:
a – comparison based on similarity thresholds; b – comparison based on nearest neighborhood [23]

RIFF is a fairly new algorithm based on the SIFT algorithm and the HOG algorithm is based on the calculation of histogram of oriented gradients RIFF resistant to image rotation. The descriptor consists of concentric annular cells applied to the image points of interest extracted by the detector of FAST (Features from Accelerated Segment Test) - algorithm for determining the points of interest in the image [19]. RIFF descriptors typically consist of four circular cells with the largest diameter of 40 pixels. In each gradient orientation ring images are calculated using the mask at the center of the derivative [-1; 0; 1] and rotated through the desired angle in accordance with the radial gradient transformation to achieve rotation invariance. The resulting gradients are quantized relative to their direction to improve performance. In addition, a local polar support frame is created in each pixel to describe the gradient from the radial and tangential directions of the center of the pixel relative to the center of the descriptor. The coordinates of the gradient in the local frame are rotation-invariant for a given descriptor center. Computational performance is further improved by sampling with a sparse gradient.

The use of RIFF tracking helps to perform the extraction and tracking of object descriptors in real time on a mobile device. With a buffer of past monitored functions and global affine models, more objects can be retrieved and tracked. This provides sufficient information for recognizing video content. The unification of tracking and recognition has an additional advantage ensures temporal consistency with the recognition data [15]. It can be concluded about the reliability of object descriptors by examining their path in a video stream. It is also worth noting that RIFF and SIFT use the same Gaussian difference to detect key points while SURF uses Hessian matrices. Comparison of the results is shown in Fig. 5, where the average number of coincidences of objects is plotted, depending on the number of turns, averaged over 10 pairs of images, where the RIFF-Annuli descriptor is the RIFF modification for half-tone images described in [15]

From these results, it can be seen that the detector of key points based on the difference of Gaussians (1) is almost isotropic which leads to a fast response for all schemes using it.

Conclusion: The RIFF descriptor is analogous to SURF, it's rotation-invariant and faster than the SIFT algorithm, which is suitable for developing augmented reality applications that work with video streaming. However, it is more prone to errors with large images, and that is why it is more often used in adapted improved using other algorithms.

RF algorithm *(Random Forest)*. The main idea is to train the recognition system based on the collected statistics of decision distribution by building a random trees forest [16]. At the stage of detection the estimated key point is “thrown”, as it were, over the output tree which is built for each image of a recognizable object taken from different angles and under different conditions. At each node the value of the probability that this reference point belongs to one of the already known ones that were defined earlier is calculated.

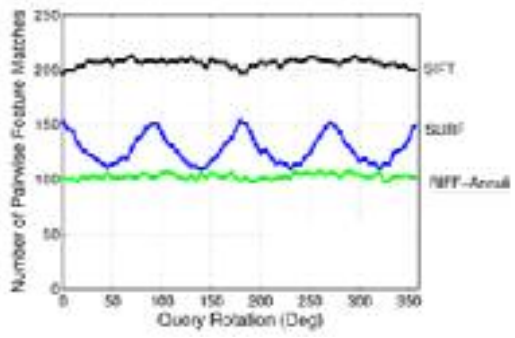


Fig. 5. Pairwise comparison at different image rotations [15]

ISSN 2663-0176 (Print)
ISSN 2663-7731(Online)

The process is performed for a set of trees and the final probability accumulates for each hypothesis that a key point belongs to a set of known points of a recognizable object. The highest probability will show the connection of the image of the original object with the reference images.

The technique belongs to the class of algorithms for selecting a neighborhood around a key point for its fixation on an image of an object. It is based on the idea of the Bayes classifier (if the density of distribution of each of the classes is known, then the desired algorithm can be represented in an explicit analytical form. Moreover, this algorithm is optimal, that is, it has the minimum probability of errors).

Let *(x,y)* be the coordinates of some point p_i in the image of the object with $p_i \in \{P\}$, where P – is the set of key points. Let $I(x_i,y_i)$ be the color value of a pixel at an arbitrary point $p_i(x_i,y_i)$, with

$$0 \le x_i < W_i \text{ and } 0 \le y_i < H_i, \quad (3)$$

where W_i and H_i, respectively, the width and height of the image in pixels.

To reduce the size of the image, it is initially reduced to monochrome based on the grayscale. In order to determine and fix a neighborhood of key points, it is necessary to carry out processing along the directions of brightness gradients around it. Just a random selection of differences in brightness gradients for an arbitrary environment of a point that gives a stable result.

The magnitude of the gradient is calculated by the formula:

$$G(x,y) = \sqrt{(I(x+1,y) - I(x-1,y))^2 + (I(x,y+1) - I(x,y-1))^2}. \quad (4)$$

Before choosing a difference, it is necessary to determine the set of vectors D_{xy}, each of which represents the values of the differences for each point.

Let $T(D_{xy})$ be a set of tests to determine the difference of the gradient on the set of points P. Test returns a bit value of 0 or 1 when the condition at the first p_1 and second p_2 points of the neighborhood in each direction is met. The result of the execution of many tests will be a bitonal mask of the direction of the gradients around the key point. The mask is a binary decision tree, in which each element of the set D_{xy} returns considered as a tree node. The transition condition "right" or "left" is determined by what value the test $T(D_{xy})$ returns when the condition at the first and second point $p_i(x_i,y_i)$ of the neighborhood is met.

Each top of the tree is a counter. The transition through the decision tree occurs with an increase in the counter by 1 according to random differences of gradients D_{xy}. As a result, after determining the set of images of the neighborhood of the same key point for different images of the same object on the set of tree vertices we obtain the probability distribution of the original image to its representation.

The search is performed for all trees in the forest corresponding to a set of reference images of the original object, and the final probability accumulates over each recognition. The maximum probability for a set of trees shows the connection of the image of the contour of the original object and the set of its images obtained at the training stage under various shooting conditions. Thus, each decision tree is built to compare the original object with a specific reference image from the database and is a decision forest for multiple images. n general, the result of building a forest is a set of F – trees, with an array of P vertices of dimension $2^{[p_{xy}]}$, where are stored the probabilities of the distribution of solutions for a particular tree *i,* to reduce the dimension and obtain independent results in recognition, so that a random subset of F_i can be selected, trees, which will include a subset of randomly selected differences D_{xy}.

At the learning stage for each hypothetical point p^* find the index of the vertex for each vector $D_{xy} \in F_i$ can be found. Then P_a represents a subset of key points for which the final probability is calculated from the tree vertices for each new point from the learning collection. As a result, we obtain the probability that the selected point p^* corresponds to some point, where P_a^i is the probability that $p^*=p^i$. The threshold number of correspondence probabilities close to 100% for the set of singular points of the image of the original object $p_i^*=p_j^i$ confirms the hypothesis that this image of the object corresponds to a certain reference image that is in the database and, therefore, this image can also be considered an image of a specific object. Thus, after each correct recognition and if the total probability of matching all key points of the original object to a particular image is less than 100% the recognized, object is considered to be the reference image obtained under the new shooting conditions and can be added to the database, which is learning the recognition systems. With the next recognition, a new decision tree is built.

However, with this approach, there is always the possibility that the set of key points of the image of the original object will always show that they correspond to the set of key points of the images of

this object from the training set. Therefore, it is necessary to perform appropriate filtering for a given priori threshold value of the probability of conformity. Since at the initial stage of learning there is no large number of images of key points for a set of images of an object. It is possible to synthesize a set of synthetic images of a particular object using various affine transformations (scaling, transfer, rotation) and a random set of noise for image distortion.

Conclusion: RF algorithm demonstrates quite acceptable quality of work even on objects with a small amount of key points (about 200). At the same time, on images with a very large number of key points, the recognition quality is rather high and the time is acceptable, which allows using the RF algorithm for systems operating in real time and on mobile devices. The algorithm is insensitive to scaling, well handles continuous and discrete features. The problem is a large amount of RAM spent during the operation of the algorithm, which in turn is critical for the implementation of the algorithm on mobile devices.

MSER algorithm *(Maximally Stable Extremal Regions)*. MSER is based on the idea of choosing regions that remain practically the same over a wide range of thresholds [17]. All pixels below the specified threshold are "black", and all pixels above or equal are "white". For the original image, if to generate a sequence of threshold resultant images of I_t where each image t corresponds to an increasing threshold t, it is possible to see a white image, then black spots appear, corresponding to the minima of the local intensity, which then increase. These "black" spots eventually emerge until the whole image turns black. The set of all connected components in a sequence is the set of all external regions (key points). In this sense, the MSER concept is associated with one of the components of the image tree. The component tree does provide an easy way to implement MSER.

Extreme areas in this context have two important properties that indicate that the set is limited:

− continuous transformation of image coordinates. This means that the algorithm is affine-invariant, and it does not matter whether the image is distorted;

− monotone transformation of the image intensity. The approach is sensitive to natural light effects, such as changing daylight or moving shadows.

Since regions (key points) are determined by the intensity function in the region and at the outer border, this leads to many key characteristics of the areas that make them useful. With a large range of thresholds, local binarization is stable in certain regions and has the properties listed below [12].

− invariance to affine transformation of image intensity;

− stability: only those regions are selected whose support is almost the same in the range of thresholds;

− detection at different scales without any smoothing, both thin and large structure is detected. However, it is worth noting that MSER detection in the pyramid of scale improves the repeatability and the number of matches in varying scales;

− the set of all external regions can be listed in the worst case with the speed $O(n)$, where n is the number of pixels in the image.

Conclusion: MSER offers the greatest variety, detects a large number of areas regardless of noise, scale and distortion force, it detects many small regions (key points) while as a rule the recognized of the large areas with a large number of errors. MSER is the most sensitive to image blur. However, multi-resolution detection allows improved blur detection.

Thus, to create a database of specialized AO markers modern algorithms were considered. They are designed to build descriptors for key points that are used for image recognition in the field of augmented reality applications. For descriptors constructing such properties of the algorithms as resistance to such interferences as changes in scale (approximation / removal), angle of capture (rotation), blurring (noise) and different illumination when acquiring the original image of an AO are important. Taking into account the analysis of the algorithms for extracting key points and the practical experience of the authors in the SDK Vuforia and Unity3D [24; 25; 26], a table of integral expert assessments of their resistance to interference is built when using them to build additional modules of the mobile application digital guide (Tabl. 1). Figure 6 shows an example of the allocation of key points for AO "Odessa Theatre of Opera and Ballet" using the SDK Vuforia and Unity3D.

Using a stack of ordered processing algorithms to build key points in the creation of a database of special marker images of AO

In addition to the estimates of the robustness of recognition algorithms to interference in obtaining the original image of an AO when creating a local version of a mobile application, digital guide should also consider characteristics, such as using a small amount of RAM and sufficient processing speed provided that the video camera even on modern

ISSN 2663-0176 (Print)
ISSN 2663-7731(Online)

smartphones does not have a high resolution which may lead to additional "blur" of the image (Tabl. 1). However, the most important characteristic is the stable recognition of AO when comparing the obtained descriptors of the original images with the reference stored in a pre-built database of specialized markers. To ensure stable recognition of AO according to the developed methodology (see Fig. 1) based on machine learning for processing architectural objects with the aim of obtaining descriptors of key points, it was proposed to create an additional module using an ordered stack (see Table 1) of the next algorithms, from the first to the last one: 1. SIFT; 2. MSER; 3. SURF; 4. RIFF; 5. RF. At the same time, the launch sequence and the number of algorithms can be changed by the user. An additional module is used to support two modes of mobile application operation, creating a database of specialized AO markers and recognizing the AO image entered by a user in order to demonstrate the corresponding AR. If the recognition mode is running, then each next stack algorithm is launched in case of the previous completion failure. The processing process can be interrupted at the request of the user or using timer settings. And the local database of specialized markers for each image except for one AR-block, which is associated with this image, simultaneously contains five types of key-point descriptors.

Conclusion and prospects for further investigation

Thus, the investigation proposed a method of recognizing architectural objects when creating mobile applications of augmented reality-type digital guide based on building a database of specialized markers. The technique consists of the following steps: processing images of architectural objects with the aim of identifying control points, obtaining descriptions of selected control points as descriptors, creating AR-metadata that correspond to architectural objects, organizing joint storage in the local database of descriptors and their corresponding metadata, visualizing the architectural object and AR metadata. For the implementation of the stages of processing images of architectural objects and obtaining descriptors of reference points, algorithms for the selection of reference points on images, such as SIFT, MSER, SURF, RIFF, RF, are analyzed (Table 1).

It is shown, that:

Table 1. Expert estimates of the sustainability to interference, resource intensity and performance in AO recognizing

Algorithm name	The value of the expert estimates of the sustainability (1-5 points)				Points		Total Σ
	Scale	Angle of rotation	Illumination	Noisy	Resource intensity	Speed performance	
SIFT	5	4	5	5	3	4	26
SIFT-PCA	5	4	5	3	5	3	26
GLOH	5	4	5	3	5	4	26
SURF	4	5	2	4	4	5	24
RIFF(*Annuli*)	3	4	4	3	5	4	23
RF	5	4	3	2	1	4	19
MSER	4	4	5	4	4	4	25

a b

Fig. 6. Example of the allocation of key points for AO "Odessa Theatre of Opera and Ballet" using the SDK Vuforia and Unity3D:
a – original image; b – image with printed key points

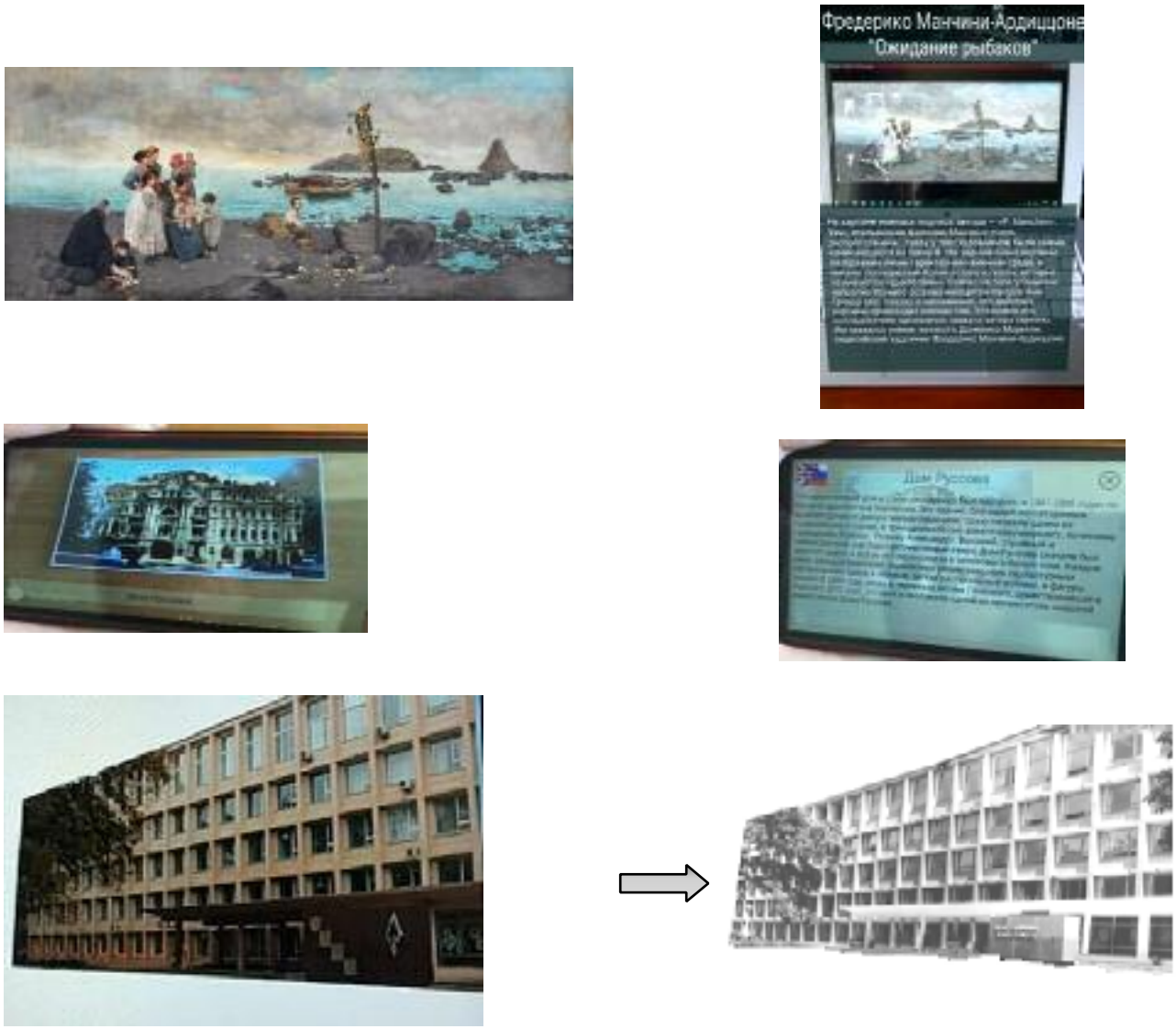
1) SIFT is resistant to approximation or removal when you hover a camera at an AO, changes in light or camera resolution but still copes with changes in the shooting angle, but at the same time it is the most resource-intensive Surf is ahead of other algorithms in speed and resistance to turning, but much more sensitive to changes in light.

2) MSER has almost all SIFT scores 1 point lower but it requires fewer resources when processing.

3) SURF is ahead of other algorithms in speed and resistance to rotation, but much more sensitive to changes in light.

4) RF with significant sustainability to approximation or removal when the camera is aimed at the AO is the most resource-intensive algorithm

Considering the above characteristics, a comprehensive sequential use of the listed algorithms for processing architectural objects with the aim of obtaining key points descriptors has been proposed for developing a digital tourist guide (Fig. 7).



a b

Fig. 7. Examples of approbation of a mobile digital guide implemented using the proposed methodology: a – original image; b – output image with AR block

ISSN 2663-0176 (Print)
ISSN 2663-7731(Online)

During the practical use of such mobile applications, it was experimentally established that the sufficient threshold is Re = 75 % (see Fig. 1) in the similarity is between key descriptors of the points of the original image and descriptors from the database of specialized markers.

References

1. Steve Aukstakalnis. (2016). "Practical Augmented Reality: A Guide to the Technologies, Applications, and Human Factors for AR and VR", Addison-Wesley Professional, 448 p.

2. Bischof, D., Droste, M., Letellier, J., Schöbinger, S., Sieck, Jü., & Thielen, E. (2018). "Development of Mixed Reality Applications for Culture and Tourism", *VI Ukrainian-German conference "Informatics. Culture. Technology"* Odessa, 12.09-22.09.18, pp. 13-20, ISBN 978-3-86488-128-2.

3. Ivanova, A. (2018). Tehnologii virtualnoy i dopolnennoy realnosti: vozmozhnosti i prepyatstviya primeneniya, [VR & AR TECHNOLOGIES: OPPORTUNITIES AND APPLICATION OBSTACLES], *Strategic decisions and risk management,* Vol. 3, pp. 88-107, ISSN 2618-947X. DOI: https://doi.org/10.17747/2078-8886-2018-3-88-107 (in Russian).

4. Yakovlev, B. S., Pusto, S. I., (2013). Klassifikatsiya i perspektivnyie napravleniya ispolzovaniya tehnologii dopolnennoy realnosti, [Classification and promising directions for the use of augmented reality technology], News of TSU, *Technical science*, No. 3, pp. 484-492 (in Russian).

5. Kipper, Greg, & Rampolla, Joseph. (2012). "Augmented Reality: An Emerging Technologies Guide to AR", *Elsevier*, 208 p. ISBN 9781597497343. Elsevier.

6. Milgram, Paul & Kishino, Fumio. (1994). "A Taxonomy of Mixed Reality Visual Displays", *IEICE Trans. Information Systems*, Vol. E77-D, No. 12, pp. 1321-1329.

7. Tony Lindeberg. (2012). "Scale Invariant Feature Transform", Scholarpedia, 7(5):10491, [Electronic Resource]. – Access mode: http://www.scholarpedia.org/article/Scale_Invariant_Feature_Transform /. – Active link – 05.04.2019.

8. Khan, N. Y., McCane, B., & Wyvill, G. (2013). "SIFT and SURF Performance Evaluation Against Various Image Deformations on Benchmark Dataset", Proceedings of International Conference of Digital Image Computing Techniques and Applications, [Electronic Resource]. – Access mode : http://www.cs.otago.ac.nz/staffpriv/mccane/publications/nabeel2011Sift.pdf – Active link – 25.03.2019.

9. Kirchner, M. R. (2016). "Automatic thresholding of SIFT descriptors", *IEEE International Conference on Image Processing (ICIP),* pp. 291-295, [Electronic Resource]. – Access mode : https://www.semanticscholar.org/paper/Automatic-thresholding-of-SIFT-descriptors-Kirchner/e9af96d478b487fec9a06dde9e43b2ed3355ea7b– Active link – 05.01.2019.

10. Gabriel, Cristóbal, Lauren,t Perrinet, Matthias, & S. Keil, (2018). "Biologically Inspired Computer Vision: Fundamentals and Applications". DOI: 10.1002/9783527680863, [Electronic Resource]. – Access mode : https://onlinelibrary.wiley.com/doi/book/10.1002/9783527680863. – Active link – 05.12.2018.

11. Pawlak, Zdzisław. (1991). "Rough Sets: Theoretical Aspects of Reasoning About Data". DOI: 10.1007/978-94-011-3534-4, [Electronic Resource]. – Access mode : https://www.researchgate.net/publication/44929167_Rough_Sets_Theoretical_Aspects_Of_Reasoning_About_Data – Active link – 01.01.2019.

12. Gavrila, D. M., Giebel, J., & Munder, S. (2004). "Vision-based pedestrian detection: the protector system", *Proceedings of the IEEE Intelligent Vehicles Symposium*, Parma, Italy, pp. 13-18.

13. Kalal, Z., Matas, J., & Mikolajczyk, K. (2010). "Forward-backward error: automatic detection of tracking failures ICPR'10", pp. 2756-2759.

14. Edouard, Oyallon, & Julien, Rabin. (2015). "An Analysis and Implementation of the SURF Method, and its Comparison to SIFT, Edouard Oyallon, Julien Rabin", *Image Processing On Line*, Vol. 5, pp. 176-218.

15. Gabriel, Takacs, Vijay, Chandrasekhar, Sam, Tsai, David, Chen, Radek, Grzeszczuk, & Bernd, Girod. (2013). "Rotation-invariant fast features for large-scale recognition and real-time tracking", Image Commun. 28, 4 (April 2013), pp. 334-344. DOI=http://dx.doi.org/10.1016/j.image.2012.11.004.

16. Denisko, D., & Hoffman, M. M. (2018). "Classification and interaction in random forests", *Proceedings of the National Academy of Sciences of the United States of America,* 115(8), pp. 1690-1692, doi:10.1073/pnas.1800256115.

17. Chavez, Aaron, & Gustafson, David. (2011). "Color-Based Extensions to MSERs", Isvc. pp. 358-366.

18. Lindeberg, Tony. (2015). "Image matching using generalized scale-space interest points", *Journal of Mathematical Imaging and Vision,* Vol. 52, No. 1, pp. 3-36.

19. Edward, Rosten, Tom, Drummond. (2006). "Machine learning for high-speed corner detection, *Proceedings of the 9th European conference on Computer Vision,* Graz, Austria, May 07-13, 2006, doi: 10.1007/11744023_34.

20. Shi, Cunzhao, Wang, Chunheng; Xiao, Baihua & Gao, Song. (2012). "Scene Text Detection Using Graph Model Built Upon Maximally Stable Extremal Regions", Pattern Recognition Letters. 34 (2), pp. 107-116, doi: 10.1016/j.patrec.2012.09.019 from 19.09.2012.

21. Chandrasekhar, V., Gabriel, Takacs, D. Chen, S. Tsai, R. Grzeszczuk, & B. Girod, "CHoG: Compressed Histogram of Gradients – A low bit rate descriptor", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Miami, June 2009*, [Electronic Resource]. – Access mode : .http://vijaychan.github.io/Publications/2009%20-%20Compressed%20Histogram%20of%20Gradients%20%20A%20low%20bit%20rate%20descriptor.pdf – Active link – 01.02.2019.

22. Prihodko, Irina, Chernienko, Ekaterina, Troyanovskaya, Yuliya, & Droste, Michael. (2018). Razrabotka metodiki sozdaniya AR-putevoditeley, [Development of methods for creating AR-guides], *Materials of the 3rd International Conference Project, Program, Portfolio Management,* P3M-2018, Book 3, pp. 57-60, ISSN: 2522-9435 (in Russian).

23. Herbert, Bay, Andreas, Ess, Tinne, Tuytelaars, & Luc Van Gool. (2019). "Speeded-Up Robust Features (SURF)", Preprint ETH Zurich, Katholieke Universiteit Leuven, [Electronic Resource]. – Access mode: https://www.vision.ee.ethz.ch/~surf /eccv06.pdf – Active link – 11.03.2019.

24. Dermenzhi, D. P., Uzun, I. S., Troyanovskaya, Yu. L. (2019). Issledovanie tehnologii raspoznavaniya dvumernyih markerov dopolnennoy realnosti na osnove freymvorka VUFORIA, [Research of recognition technology of two-dimensional markers of augmented reality based on VUFORIA framework], *Materials of the Eight International Education Conference of Students and Young People in the International Information Technologies 2019, 23-25 March 2019* / MES of Ukraine; Odessa National Polytechnic University, Institute of Computer Systems, pp.149-150 (in Russian).

25. (2019). "Vuforia developer", [Electronic Resource]. – Access mode : https://developer.vuforia.com/. – Active link – 01.02.2019.

26.(2019). "Unity3D", [Electronic Resource]. – Access mode : – https://unity3d.com/ru - Active link – 01.02.2019.

Received 15.03.2019

УДК 004.9

[1]**Арсірій, Олена Олександрівна**, доктор технічних наук, професор, зав. кафедри інформаційних систем інституту комп'ютерних систем, E-mail: e.arsiriy@gmail.com, ORCID: 0000-0001-8130-9613
[1]**Трояновська, Юлія Людвигівна**, старший викладач кафедри інформаційних систем інституту комп'ютерних систем, E-mail: troyanovskaja@opu.ua, ORCID: 0000-0002-6716-9391
[1]**Приходько, Ірина Олександрівна**, магістр кафедри інформаційних систем інституту комп'ютерних систем, E-mail: ir.prikhodko@gmail.com, ORCID: 0000-0002-7846-9867
[1]**Котова, Діана Юріївна**, бакалавр кафедри інформаційних систем інституту комп'ютерних систем, E-mail: kotowstrider@gmail.com, ORCID: 0000-0002-1067-0433
[1]Одеський національний політехнічний університет, пр. Шевченка, 1, Одеса, Україна, 65044

МЕТОДИКА РОЗПІЗНАВАННЯ АРХІТЕКТУРНИХ ОБ'ЄКТІВ В ТЕХНОЛОГІЯХ ДОПОВНЕНОЇ РЕАЛЬНОСТІ НА ОСНОВІ ПОБУДОВИ БАЗИ СПЕЦІАЛІЗОВАНИХ МАРКЕРІВ

Анотація. *В роботі запропонована методика розпізнання архітектурних об'єктів при створенні мобільних додатків доповненої реальності на основі побудови бази спеціалізованих маркерів. На основі аналізу методів технології доповненої реальності для розпізнавання архітектурних об'єктів був обраний метод, заснований на спеціальних маркерах. Виділено ряд алгоритмів розпізнавання образів, що підходять для даного завдання. Це алгоритми, засновані на виборі ключових точок зображень і їх дескрипторів. Основною метою роботи є створення методики, яка забезпечує стабільне*

розпізнавання архітектурних об'єктів в мобільних додатках для створення цифрового гіда доповненої реальності на основі спеціалізованих маркерів. Науковою основою дослідження є системний підхід при аналізі розглянутих алгоритмів розпізнавання маркерів, використовуються машинне навчання для розробки бази даних зображень маркерів і розпізнавання АТ. Методика складається з наступних етапів: обробка зображень архітектурних об'єктів з метою виділення опорних точок, отримання опису виділених опорних точок у вигляді дескрипторів, створення AR-метаданих, які відповідають архітектурним об'єктам, організація спільного зберігання в локальній базі дескрипторів і відповідних їм метаданих, візуалізація архітектурного об'єкта і AR-метаданих. Для реалізації етапів обробки зображень архітектурних об'єктів і отримання дескрипторів опорних точок, проаналізовані алгоритми виділення опорних точок на зображеннях, такі як SIFT, MSER, SURF, RIFF, RF. Показано, що дані алгоритми є інваріантними до масштабування, обертання, а також стійкими до змін освітленості, шуму і кута перегляду. Запропоновано комплексне їх використання для обробки архітектурних об'єктів з метою отримання дескрипторів опорних точок. Для забезпечення стабільного розпізнавання АТ відповідно до розробленої методики, заснованої на машинному навчанні для обробки архітектурних об'єктів з метою отримання дескрипторів ключових точок, було запропоновано створити додатковий модуль з використанням упорядкованого стека алгоритмів, в якому послідовність запуску і кількість алгоритмів можуть бути змінені.

Ключові слова: *доповнена реальність; AR-технології; маркерні методи розпізнавання*

УДК 004.9

[1]**Арсирий, Елена Александровна**, доктор технических наук, профессор, зав. кафедры информационных системинститута компьютерных систем, E-mail: e.arsiriy@gmail.com, ORCID: 0000-0001-8130-9613

[1]**Трояновская, Юлия Людвиговна,** старший преподаватель кафедры информационных систем института компьютерных систем, E-mail: troyanovskaja@gmail.com, ORCID: 0000-0002-6716-9391

[1]**Приходько, Ірина Олександрівна**, магистр кафедры информационных систем института компьютерных систем, E-mail: ir.prikhodko@gmail.com, ORCID: 0000-0003-0926-7185

[1]**Котова, Диана Юриевна**, бакалавр кафедры информационных систем института компьютерных систем, E-mail: ir.prikhodko@gmail.com,ORCID: 0000-0003-0926-7185

[1]Одесский национальный политехнический университет, пр. Шевченко, 1, Одесса, Украина, 65044

МЕТОДИКА РАСПОЗНАВАНИЯ АРХИТЕКТУРНЫХ ОБЪЕКТОВ В ТЕХНОЛОГИЯХ ДОПОЛНЕННОЙ РЕАЛЬНОСТИ НА ОСНОВЕ ПОСТРОЕНИЯ БАЗЫ СПЕЦИАЛИЗИРОВАННЫХ МАРКЕРОВ

Аннотация. *В работе предложена методика распознания архитектурных объектов при создании мобильных приложений дополненной реальности на основе построения базы специализированных маркеров. На основе анализа методов технологии дополненной реальности для распознавания архитектурных объектов был выбран метод, основанный на специальных маркерах. Выделен ряд алгоритмов распознавания образов, подходящих для данной задачи. Это алгоритмы, основанные на выборе ключевых точек изображений и их дескрипторов. Основной целью работы является создание методики, обеспечивающей стабильное распознавание архитектурных объектов в мобильных приложениях для создания цифрового гида дополненной реальности на основе специализированных маркеров. Научной основой исследования является системный подход при анализе рассмотренных алгоритмов распознавания маркеров, используются машинное обучение для разработки базы данных изображений маркеров и распознавания АО. Методика состоит из следующих этапов: обработка изображений архитектурных объектов с целью выделения опорных точек, получение описания выделенных опорных точек в виде дескрипторов, создание AR-метаданных, которые соответствуют архитектурным объектам, организация совместного хранения в локальной базе дескрипторов и соответствующих им метаданных, визуализация архитектурного объекта и AR-метаданных. Для реализации этапов обработки изображений архитектурных объектов и получения дескрипторов опорных точек, проанализированы алгоритмы выделения опорных точек на изображениях, такие как SIFT, MSER, SURF, RIFF, RF. Показано, что данные алгоритмы являются инвариантными к масштабированию, вращению, а также устойчивыми к изменениям освещенности, шума и угла просмотра. Предложено комплексное их использование для обработки архитектурных объектов с целью получения дескрипторов опорных точек. Для обеспечения стабильного распознавания АО в соответствии с разработанной методикой, основанной на машинном обучении для обработки архитектурных объектов с целью получения дескрипторов ключевых точек, было предложено создать дополнительный модуль с использованием упорядоченного стека алгоритмов, в котором последовательность запуска и количество алгоритмов могут быть изменены.*

Ключевые слова: *дополненная реальность; AR-технологии; маркерные методы распознавания*