Compression and Transmission of Video Stream over Mesh Network

Sergey A. Kuzmin, Ryfkat R. Bulyakulov
Saint Petersburg State University of Aerospace Instrumentation
Saint Petersburg, Russia
kuzmin_serg@list.ru, alien109@mail.ru

***Abstract*—The article is devoted to the tasks of video compression and video transmission for surveillance system by means of effective decreasing of redundancy. The conception of codec is a combination of layered representation and differential pulse-code modulation. The layers, from which each frame of decoded video is composed, are alpha channel, background estimation, image with difference between intensities of objects and background estimation, correcting image. Layers are compressed with WebP. The result of test on real videos shows that proposed codec outperforms MPEG4 and H.264 in the desired range of values of compression factor. The reliable transmission of video over long distance is a complex task, which was solved by the use of mesh network.**

I. INTRODUCTION

One of emerging applications of mesh networks is video transmission [1]. The purpose of this article is to provide information about video codec and system for video transmission, whose are developed by authors in SUAI.

There are exists two types of mesh networks by hardware (wired and wireless) and two types of mesh networks by topology (true and partial). We would describe our view of using partial mesh network with bridging between wireless and wired segments.

The structure of article is as follows: at first we describe general information about video codec, at second we describe segmentation of video objects in video codec, at third we describe transmission in mesh network.

The source video is produced by traffic surveillance system. The problems, which are need to be solved: 1). development of ad hoc video codec, which would utilize the specific structure of frames in video (static camera, static background, moving objects) and wouldn't have big peak-to-average ratio of bit rate. 2). transmission of compressed video over mesh network to desired server through several intermediate nodes, whose are selected among possible nodes by optimization of some cost function.

II. VIDEO CODEC

The existing codecs works in a very wide range of values of compression factor, but still have some weak points: 1) limited work with non standard resolutions, 2) orientation on rapid changes in video like in news on TV, 3) Groups of Pictures approach leads to delays in work of codec and the necessity of big buffers, 4) problem of crest factor (peak-to-average ratio) of bit rate in case of Groups of Pictures approach, 5) no support for metadata or limited options for annotating individual frames, which is an essential limit in case of video surveillance, because it is common practice to create inverted index for fast search of objects [2].

Problem of video coding in case, if we don't decrease duration, can be formulated in the this way: if we denote C – channel capacity, R_S – bit rate of the source file, R_C – desired bit rate of compressed file, then the general task is to make

$$R_C \le C.$$

The problem is that compression factor $K=R_S/R_C$ is usually equal to several dozens or even hundreds. The rational approach to compression is based on reduction of redundancy. The main types of redundancy in this case: statistical (eliminated by lossless compression) and psychovisual (eliminated by lossy compression). Let us denote $K=K1\times K2$ as a product of compression factors of two types of approaches – $K1$ from lossy and $K2$ from lossless compression. Both of these redundancies have spatial ($K1_S$, $K2_S$) and temporal ($K1_T$, $K2_T$) components.

Let us show the difference between transmission in wired and wireless segments. The main difference is in required value of K. The bit rate of the source file is a volume-to-duration fraction $R_S=V/T$. The volume depends on quantity of pixels in frame S, bits per color channel bpc, frequency F, quantity of color channels P and duration T

$$V=S\times bpc\times F\times P\times T.$$

In case of wired transmission R_S is computed as

$$Rs=S\times bpc\times F\times P.$$

In case of wireless transmission the formula from Shannon–Hartley theorem has been used, which indicated the need to amplify signal in transmitter

$$R_{SA}=B\times\log_2(1+SNR),$$

where B – bandwidth of the channel, SNR – required signal-to-noise ratio for correct transmission.

The upper bound estimate of B can be derived in form $B=S\times F\times P/2$ and the estimate of SNR can be derived in form $SNR=r^2\times B^2/(16\times\pi^2\times G1\times G2\times c^2)$, where $G1$ and $G2$ – gains of

the antennas (transmitter and receiver), c – speed of light, r – distance between antennas.

If we suppose that $R_C=C$ and write bpc as a sum of statistical redundancy D and entropy H, then we have equations for two cases with corresponding channel capacities C_{WIRED} and $C_{WIRELESS}$:

$$K1_S \times K1_T \times K2_S \times K2_T = \begin{cases} \dfrac{2 \times B \times (D+H)}{C_{WIRED}}, wired \\ \dfrac{B \times \log_2(1+SNR)}{C_{WIRELESS}}, wireless \end{cases}.$$

It should be mentioned that formula SNR=6,02*bpc+10,8 dB is linking bpc and SNR in wired case. As it seen from these formulas, wireless transmission usually requires bigger compression factors at long distances, than in wired case. The redundancy D is related to compression factor $K2_S$. Next, we will focus on decreasing of bit rate Rs by decreasing of temporal redundancy (increasing $K2_T$).

The simple experiments shows that we should not decrease bpc and P as a primary means of compression. So we must analyze source video and find some data in it, which should not be (fully) transmitted in every frame. It means that certain part of S pixels should be transferred at frame rate F and the other part of frame shouldn't – this leads us to concept of adaptive rate of renewing intensity for every pixel.

Let us formulate it in this way: which parts of image S shouldn't be sent at frame rate F? The answer is: parts of image, which are the same as they were in previous frames. In many cases these parts of frame are located at background of image. So, if the intensities of background pixels Ib were constant over time, we can send them only one time. Fig.1 shows that actually big share of area of frame in typical video doesn't have big changes over time.

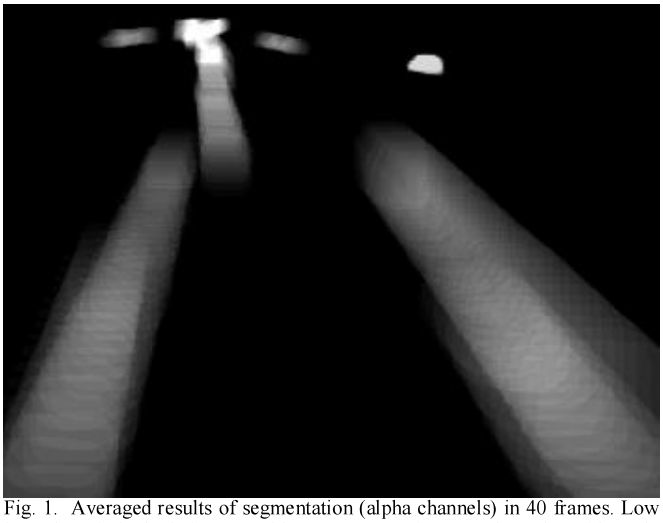

Fig. 1. Averaged results of segmentation (alpha channels) in 40 frames. Low intensity of pixel (dark parts of image) means absence of moving objects in this part.

We got two problems here: 1) how to detect background pixels? 2) parts of background change their intensities over time, oscillating around some mean value. The second problem actually give us a hint: let us calculate (weighted) mean values for period of time for intensities of pixels and use it in two ways: 1) we use calculated image (so called background estimation Ibe) to detect areas of input frames with intensities, which are different from background; 2) we can use this background estimation as a prediction of intensities of background in encoder and decoder.

In first way the absolute value of a difference between input image and background estimation is compared to threshold. Also the absolute value of difference between bitmaps with edges in input image and edges in background estimation is compared to threshold. These two thresholded images are used for formation of alpha channel A, as described in section III of article. The alpha channel A is a binary image (black color – background (A=0), white color – object (A=1)). In the second way we make synthesis of image Is in decoder from known background estimation, alpha channel and foreground layer

$$Is=Ibe + A \times (Iob-Ibe),$$

where Ibe - intensity of pixel in background estimation image, Iob – intensity of object in this pixel in input image. Everything seems fine, but we miss two things: errors in detection of class of pixels (background/object separation) and change of intensities of background. The actual formula for intensity in every pixel of input image is

$$I=Ibe + A \times (Iob-Ibe) + (1-A) \times (Ib-Ibe),$$

in case of perfect detection of classes of pixels, where Ib - intensity of background pixel in input image. Let us denote

$$Iu= I-Is,$$

as a correcting image for errors in detection of class of pixels and background estimation. It is shown on Fig.2 that the background and shadows in Fig 2b look different from Fig 2a.

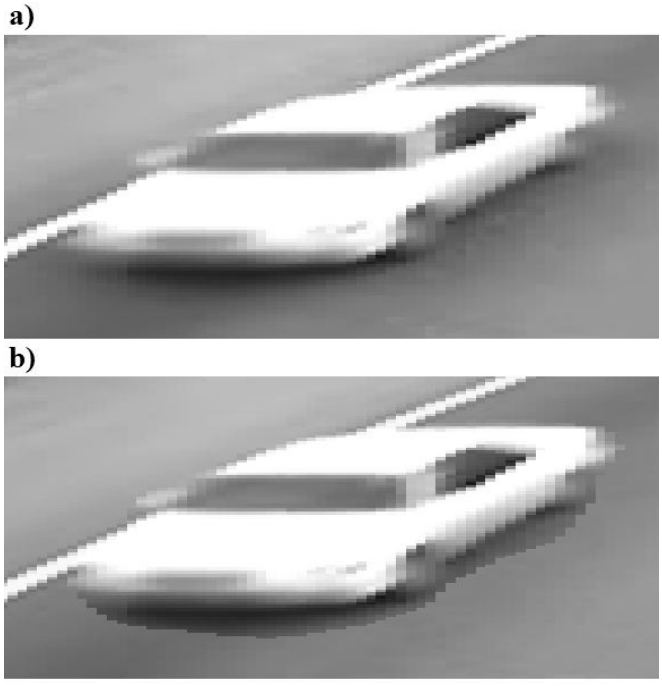

Fig. 2. Comparison of input image and image after synthesis, but before correction: a) part of current frame (source), b) part of image during decoding (after synthesis – moving car located on background estimation)

The layers are compressed with WebP codec, so some degradation of quality is happened, mainly for *Iob-Ibe* and *Iu*□ layers. The degraded versions of layers, transmitted over channel, and images, formed from degraded versions of layers, will be denoted with superscript *'*.

Finally, decoded image *I'* is a sum of result of synthesis *Is'* and correcting image *Iu'*

$$I' = Is' + Iu',$$

but it is not identical to uncompressed source frame due to mentioned compression of layers.

Codec consist of encoder and decoder. Scheme of encoder is shown on Fig.3.

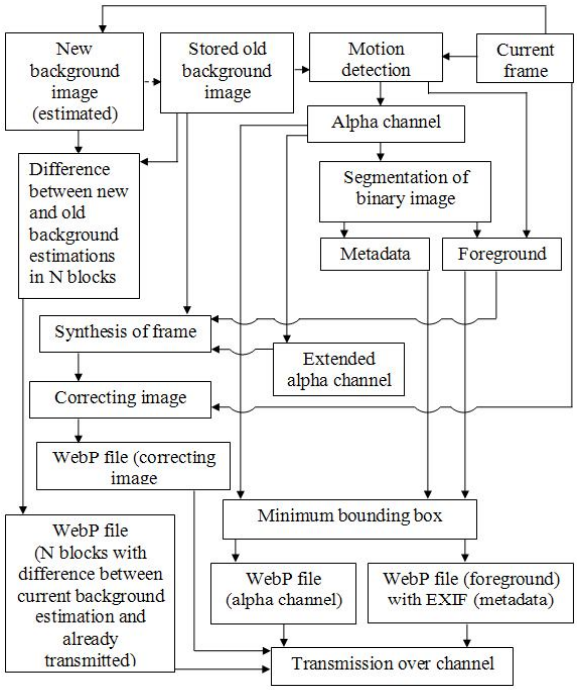

Fig.3. Scheme of encoder in proposed codec

Encoder consists of several units (or blocks): background estimation unit, segmentation (pixel classification) unit, extraction of areas with objects, synthesis unit, correcting image calculation unit, etc. Decoder is doing operations in inverse order.

The work of encoder starts in the block "Motion detection" (described in section III of article; fig. 5). Image is segmented using background estimation image. The resulted binary image plays key role in creation of image with difference between intensities of objects and background estimation (block "Foreground"). The copy of binary image is called alpha channel. Foreground, background estimation and alpha channel are mixed for creation of synthesized image. The difference between current frame and synthesized image forms correcting image.

Nature of modern image formats, based on transforms of blocks of pixels, give severe degradation of quality in decoded image. For better quality of compressed frames alpha channel is converted into extended alpha channel (if any pixel of block is white (include parts of foreground), then all pixels of this block will be white). Extended alpha channel used only for correct synthesis and calculation of correcting image inside encoder, the decoder operates with usual alpha channel as shown in scheme. The minimum bounding box is a well known technique, used for efficiency. Another sort of task is a renewing of background estimation image. It can be done at a very low speed, by sending several blocks per frame.

Let us discuss the novelty of proposed codec. The output frame is not relies on adjacent frames and this distinguishes it from Group of Pictures approach. Our approach with steps of prediction, based on background estimation, and correction reminds us a differential pulse-code modulation (DPCM) scheme [3]. The use of layers of alpha channel, background, foreground, correcting image made it similar to layered representation [4]. Layered representation is also called object-based video coding and previous researchers shows that it can outperform traditional hybrid video coding scheme of MPEG family of codecs and some forms of it already presents in MPEG 4 and other codecs [5]. But a number of differences also exists: 1. they use one correcting image, but in our approach second correcting image has been used for slow update of background; 2. they use Motion Vector Field [4], but we don't; 3. we send metadata about objects; 4. we use WebP.

The proposed codec was compared with classical codecs MPEG 4 SP (Simple profile) L6 and H.264. The experiments are examined with the famous PETS 2000 sequence from University of Reading (768x576 pixels, 1452 frames, 25 frames/s) [6].

The results of test are shown on Fig.4. This plot shows the relation between PSNR and compression factor K for different values of parameter Q (quality) in codecs. The curve of MPEG 4 SP codec was almost identical to H.264 (maximal difference between H.264 and MPEG 4 SP at the same compression factor was 0.69 dB), so it is not shown, since all the conclusions for H.264 will be valid for MPEG 4 SP.

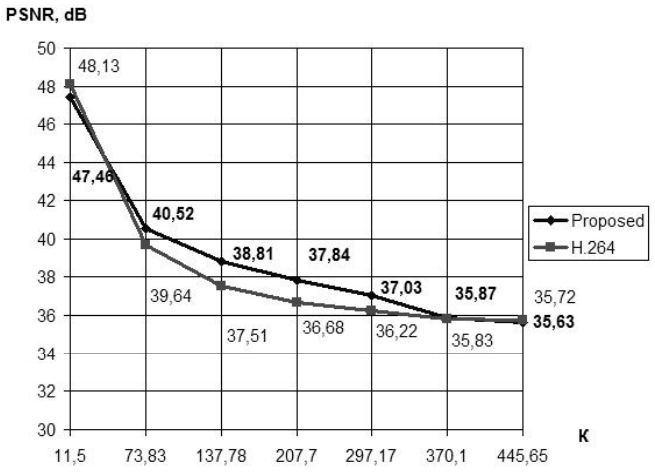

Fig. 4. Results of comparison with H.264. The bold numbers are for proposed codec and the usual numbers are for H.264 codec.

The proposed codec outperforms H.264 and MPEG 4 SP codecs in the desired range (from 40 to 390) of values of compression factor. The increment in PSNR at the same compression factor is about 0.8-1.3 dB. The increment in compression factor at the same PSNR is up to 2.15 times.

III. Segmentation of Frames and Evaluation of Performance

A. Segmentation of frames

Motion detection is carried out by combining binarized background subtraction and binarized image of difference between edges in current frame and edges in estimated backround with the OR operation (Fig. 5).

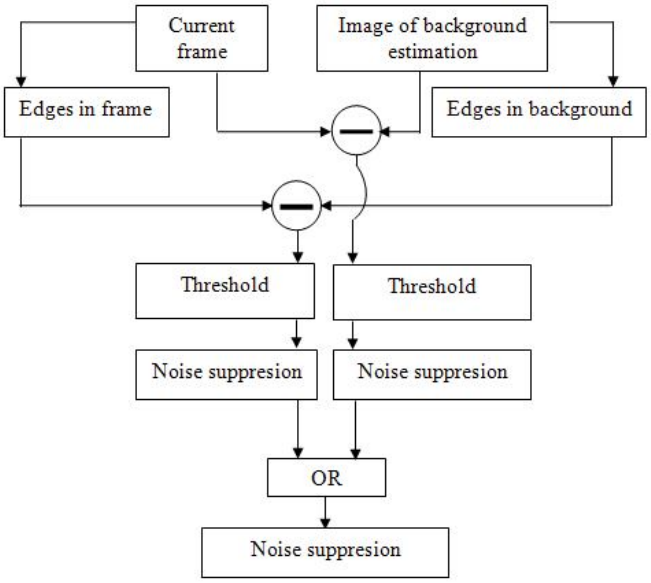

Fig.5. Scheme of motion detector, producing alpha channel

We also have more sophisticated detectors, based of wavelets, whose are detect interest pixels at higher or at lower scales, if there is a need to upscale of downscale encoded video. The idea of detection in these detectors is the same, as described earlier – combining of thresholded changes in low and high frequency bands. These wavelet based detectors and base approach, which doesn't change the scale, forms a family of motion detection algorithms, which gives a Scalable Video Coding capability to proposed codec.

Background is estimated by per pixel implementation of Kalman filter in direction of time. Binary images are filtered out by different rank filters before and after OR operation. The object detection results can be improved by suppression of shadows, which are detected along with moving objects. The thresholds are adaptive and separately calculated using cumulative distribution functions of corresponding difference images.

B. ROC curve and related concepts

In information retrieval, signal detection, remote sensing, computer vision it is common practice to use Receiver Operating Curve (ROC) for comparing algorithms.

ROC curve is a plot, showing how *TPR* (true positive rate) depends on *FPR* (false positive rate): $TPR = f(FPR)$. Concepts of *TPR* and *FPR* arise from the following example: let us consider automatic detector, observer (human) and set of data with N dimensions (3 dimensions in case of image – 2D plane of coordinates and 1D of amplitudes). We run detector over image and it classified pixels in two classes – positive (interesting parts of foreground – objects) and negative (background). The observer do the same job (make his decisions). Let's denote set of observer's decisions as real classes of pixels and set of detector's decisions as attempts. The cases in classification of one pixel are shown in Table I.

TABLE I. Cases in Classification

Real class/ Attempt	Negative	Positive
Negative	True negative	False positive
Positive	False negative	True positive

The sums of probabilities of cases are equal to 1:

$$P(True\ negative)+P(False\ positive)=1,$$
$$P(False\ negative)+P(True\ positive)=1.$$

The performance of detector depends on used algorithm of classification (value of threshold, etc.) and image itself (contrast, etc.). We get set of pairs of *TPR* and *FPR* by changing threshold and draw the plot called ROC. Here is a remarkable point in using ROC: algorithm can be successfully used for moving object detection only if *TPR* is higher than *FPR* (because $TPR = FPR$ means random choice [7], $TPR < FPR$ means that detection of background is performed instead of detection of foreground).

It is important to note that in case of absence of a priori information about object intensity values (or colours) bottom-up approaches are used in a following way: they detect interesting pixels (points with high difference with neighbours in space or time), then post-process these pixels and group them in objects. Thus here is can be different ways to evaluate performance: interesting pixels detection rate and object detection rate. In our application it is important to have both interesting pixels detection rate and object detection rate in low error zone. Interesting pixels detection rate changes during multiple-step post-processing, i.e. ROC curve can be computed after each step of processing. It is important to measure changes of ROC for choosing parameters of post-processing algorithms.

Each ROC curve consists of several working points (where $TPR = f(FPR)$ was measured). Threshold of binarization (or another parameter, which leads to change of proportion in number of white and black pixels) is changed for measurement of $TPR = f(FPR)$ in working points. Ground truth data is needed for measurement. In our case of measurement of interesting pixels detection rate it is binarized image. Decision for every pixel during binarization can be false or true. There are two type of errors: foreground (object) detected instead of background, background detected instead of foreground. First type of error lowers *TPR* and second type of error increases

FPR. Joining of algorithms and post processing steps can change intensity of initial interesting points. Therefore performance cannot be perfect, but it should be near perfect. Some limited number of errors, which do not change class of detected objects, is allowed. *TPR* and *FPR* are usually measured as probabilities, i.e. values in range from 0 to 1. It is possible to scale them in range from 0 to 100 (as percents). In space of plot $TPR = f(FPR)$ it is possible to denote zone of desired performance (ZDP) in top left corner (Fig. 6).

Any ROC, which have at least 1 point inside zone of desired performance, is denoted as desired ROC. It is possible to imagine myriads of these ROC curves, but it is hard to implement any of them. It is possible to approach ROC of real algorithm to desired ROC by boosting (joining of motion detection algorithms) and post processing.

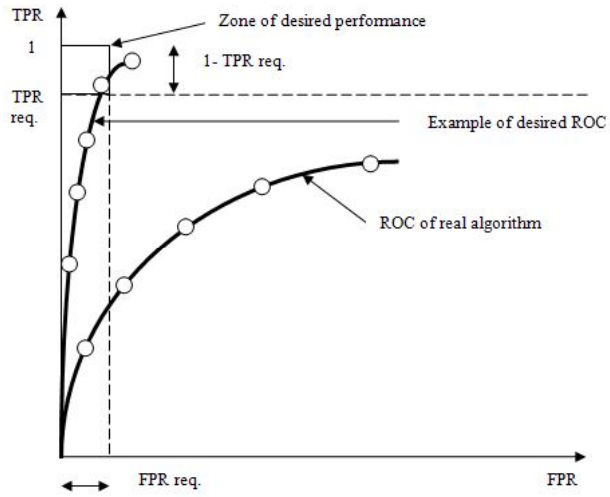

Fig. 6. Desired ROC must have at least one point in the zone of desired performance

C. Set of ROC curves in case of boosting and multiple-step processing. Path to Zone of Desired Performance.

In computer vision by joining of algorithms and multiple repeating of filtering in post processing it is possible to reduce *FPR* and increase *TPR*. ROC curves, measured at every step of processing, can be shown at one plot. Evolution of ROC curve and step-by-step progress are interesting for developers.

Fig. 7 demonstrates ROC curves of two algorithms (1 alg., 2 alg.) before joining with operation OR, resulting ROC after joining and several steps of filtering.

D. Measurement of performance of motion detection algorithms.

There are exists several ROC-based approaches for evaluation of performance:

1) value of *TPR* at reference (or allowed) *FPR*;
2) area under curve (AUC) - shows mean *TPR* value in full range of *FPR* values;
3) *M*1 and *M*2 measures, proposed by one of the authors recently [8]. The positive feature of these measures - no need to know allowed *FPR* and correct estimation of optimality.

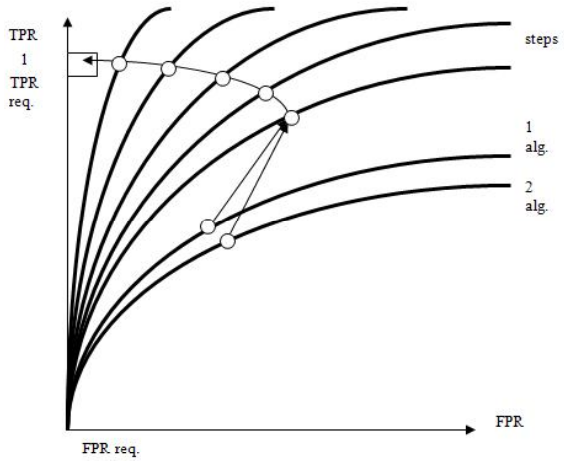

Fig. 7. ROC after boosting (joining of weak classifiers) and post processing. Path to ZDP is shown as arrows

In this work we calculate value of *TPR* at reference *FPR* and the *M*2 measure.

The performance of segmentation algorithm is very high – value of *TPR* is almost 0.99 at reference *FPR*, equal to 0.01 (Table II).

The M2 measure is relatively new and we describe its calculation in detail.

Approach is based on following speculations:

1) for best algorithm *TPR*=1 and *FPR*=0;
2) for worst algorithm *TPR*=0 and *FPR*=1;
3) for all other algorithms *TPR* is in the range between 1 and 0 and *FPR* is in the range between 0 and 1.

It is possible to write straight line equation, which describes positions of all algorithms from best down to worst:

$$TPR+FPR=1.$$

This line of algorithm's perfectness has a cross point with ROC curve. This cross point is denoted as sum100 point (1 is equal to 100 in percents).

Actually this equation is derived from Equal Error Rate criterion (value of *FPR* in point of ROC, where 1-*TPR*=*FNR*=*FPR*) in three steps:

$$FNR=FPR,$$

$$1-TPR=FPR,$$

$$TPR+FPR=1,$$

where *FNR* is a false negative rate.

These line and point have some interesting features:

1) line of algorithm's perfectness is perpendicular to random choice line *TPR*=*FPR* (which is a case of signal-to-noise ratio SNR=0);

2) line of algorithm's perfectness is connected to left top point of zone of desired performance. It seams like it is a direction of best choice;
3) let's denote, that sum100 point have a coordinates *TPR**, *FPR**. This point shows how many false positives and true positives among 100 positive decisions.

Calculation of measure M2 is shown on Fig. 8.

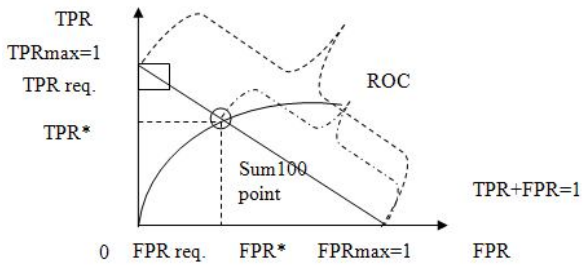

Fig. 8. Calculation of performance measure *M*2 at cross point sum100

From similarity of triangles (length of line segment (from sum100 to *FNR*=1) is divided by full line's length)

$$M2 = TPR^*/TPRmax = (1-FPR^*)/FPRmax.$$

It is possible to simplify it, because *TPRmax*=1 and *FPRmax*=1: $M2 = TPR^* = (1-FPR^*)$.

The measured working points of ROC are placed in Table II.

TABLE II. POINTS ON ROC CURVE

TPR	FPR
0.9986	0.0321
0.9937	0.0123
0.9854	0.0058
0.9788	0.0037

The *M*2 measure is equal to 0.988.

IV. TRANSMISSION IN MESH NETWORK

For such a big system as traffic surveillance system, it is essential to have reliable connection between central post and sensors. The reliability can be achieved by different approaches (special modulation, error correction codes, etc.), but one of the oldest approaches is duplication. So we need to have several alternative working lines of communication for every sensor.

It is money consuming to have wired connections in such a wide area, so we propose to use radio modems (base stations), situated along with every distant sensor, set of network bridges, arranged concentrically around the central post, and the Ethernet cables at distances, closest to central post. The Ethernet have much higher throughput, than radio modem, and since at close to central post we need to transfer much bigger stream, than at distant sensors, it is a natural choice. The example of radio modem with high speed of transmission is CalAmp Phantom II [9]. The required compression factor in this case is about 243 times for video stream with standard definition. The conditions of radio transmission are very different and can change over time, so we decided to create a mesh network of base stations. We need to choose the next base station among alternatives for every base station along the path to central post in multihop case of transmission. The main approaches are principle of shortest path [10] and utility function for maximizing the total quality [1]. We modify the second approach. The known approach to link quality estimation is measurement of SNR [11]. Base stations are sending regularly special signal of known power and duration. We calculate the link quality by averaging of several received test signals and then calculating the relative average power of received signal (dividing the average power to maximal power). It is simpler than calculation of SNR. The results of test of link quality forms a matrix (Table III).

In this example each of base stations has two alternative next stations. We consider greedy algorithm as a routing approach. The Hungarian algorithm, minimax algorithm and the alpha beta pruning can be used for more complex routing algorithms.

TABLE III. EXAMPLE OF RELATIVE AVERAGE POWERS FOR STATIONS

Base/Alternative	Station 1	Station 2	Station 3
Station 1	x	0.4	1.0
Station 2	0.3	x	0.5
Station 3	0.7	0.6	x

V. CONCLUSION

The temporal redundancy is high in big shares of areas of frames and this fact lead us to approach, similar to differential pulse-code modulation [3], combined with ideas from layered representation [4]. The novel elements were discussed in section II. Separation of input frame into several layers gives higher quality at the same compression factor (the increment is about 0.8-1.3 dB) or higher compression factor at the same PSNR (the increment is up to 2.15 times), than in H.264 and MPEG 4 SP. This result is obtained by the use of concept of adaptive rate of renewing for every pixel (actually for every block of pixels in our implementation). The implementation of concept is mainly based on reduction of temporal statistical redundancy, the other types of redundancies are decreased by WebP image codec. The compressed video is transmitted through mesh network to central post.

ACKNOWLEDGEMENT

S. A. Kuzmin thanks SUAI for personal grant PSR 3.1.2-1, which supported his research on improving algorithms of image and video processing.

REFERENCES

[1] David Q. Liu, Jason Baker, "Streaming Multimedia over Wireless Mesh Networks", *International Journal of Communications, Network and System Sciences*, 2008, vol 1, issue 2, pp. 177-186.

[2] S.A. Kuzmin, R.R. Bulyakulov, "Video Processing In Codec, Which Is Better Than H.264", *in Proc. of the 2016 IEEE North West Russia Section Young Researchers in Electrical and Electronic Engineering Conference*, 2016. Pp. 267-269.

[3] U.S. patent 2605361, C. Chapin Cutler, "Differential Quantization of Communication Signals", filed June 29, 1950, issued July 29, 1952.

[4] Wang, J. Y. A.; Adelson, E. H. "Representing moving images with layers", *IEEE Transactions on Image Processing*, vol. 3, issue 5, 1994, pp.625-638.

[5] A. Krutz, S. Knorr, M. Kunter, T. Sikora, "Camera Motion-Constraint Video Codec Selection", *in Proc. of the IEEE 10th International Workshop on Multimedia Signal Processing*, 2008. Pp.58-63.

[6] University of Reading FTP server, sequence of images for the PETS2000 workshop, filmed by Computational Vision Group. Web: ftp://ftp.pets.reading.ac.uk/pub/PETS2000/test_images/

[7] T. Fawcett. "ROC graphs: Notes and practical considerations for researchers". HP Labs Tech Report HPL-2003-4.

[8] S. A. Kuzmin, "New Performance Measures in Object Detection", *in Proc. of the 2015 IEEE North West Russia Section Young Researchers in Electrical and Electronic Engineering Conference*, 2015. Pp. 86-89.

[9] CalAmp official website, CalAmp Phantom II specifications, Web:: http://www.calamp.com/products/private-radios-narrow-band-equipment/ip-modems-and-routers/phantom-ii

[10] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network Flows: Theory, Algorithms and Applications.* Prentice Hall, 1993.

[11] D. Lal, A. Manjeshwar, F. Herrmann, E. Uysal-Biyikoglu, A. Keshavarzian, "Measurement and Characterization of Link Quality Metrics in Energy Constrained Wireless Sensor Networks", *in Proc. Globecom 2003*, pp. 446 – 452.