UDC 004.043

A. Shportko
State humanitarian university of Rivne,
department of economic cybernetics

THE USE OF PREDICTORS IS IN THE PROCESS OF MAKING PROGRESS HIERARCHICAL CONTEXT-INDEPENDENT COMPRESSION OF IMAGES WITHOUT LOSSES

Argued expedience, the method of round of pixels is resulted and the predictors for realization of progressing hierarchical context-independent compression of images without losses are offered . The results of application of proposed approaches for diminishing entropy of representing the set of ACT in the process of previous transformations are presented.

Keywords - progress compression of images, compression without losses, predictors, entropy

Entry

In the modern world images are the inalienable part of multimedia information, which is more frequent created, accumulates and saved on digital transmitters and passed by communication channels. The compression of the proper files enables to promote speed of exchange information for networks proportionally and decrease the volumes of the use of disk space. For today the analysis of brightness pixels of images in popular graphic formats, which execute a compression without losses (for example, in the format of PNG [4, p. 249-317]), is carried out consistently on lines from top to bottom, and in every line – from left to right. As a result, to show out the compressed image in these formats is possible only after decoding of all pixels. Decompression of snapshots or pictures with millions of pixels at such method of round can last a few seconds regardless of the size of area and the settling ability of device.

Next to it, for the acceleration of output of large images in the formats of compression with losses progress hierarchical working of pixels is used more frequent [6, p. 176]. In the process of application of this method of round of image is worked out over layer, multiplying settling ability each time, thus in the process of the sequential processing of data of the next layer information of previous layers is used. An image from the pixels of duty layer actually is the diminished copy of image from the pixels of next layer, and the last layer coincides with an entrance image. That is why during the progressing hierarchical decoding the details of the image show up gradually. To stop such decoding is possible already after decompression of the layer with the amount of pixels, not smaller than the area of conclusion on each axe, not expecting the recreation of all pixels of the image. Consequently, the **development of the methods and graphic format of the compression of images without losses with the use of principles of the progressing hierarchical analysis**, which is a purpose of research, **is an actual task for today**.

The analysis of the last researches and publications. The principles of compression of images without losses with the use of predictors

As it is known, compression of images without losses in graphic formats is done more often in three steps: on the first step the brightness of pixels is transformed by predictors; on the second a context-dependent code diminishes surplus between similar fragments; on third context-independent code removes surplus between the prevailing values of brightness components. A context-dependent code can diminish an aspect of compression (attitude of sizes of the compressed toward uncompressed files of image, in future – AC) ratio in a few times owing to similar fragments. But such fragments are rarely happened in photorealistic images, that is why the unique universal stage of compression of images without losses is a context-independent code. Basic principle of such code is that the **length of code of arbitrary element with greater probability must not exceed the length of code of any element with less probability.** This principle is based on fundamental position of information theory. Based on it for minimization of length of code of sequence element s_i (in our case an element is a brightness of separate component of every pixels) with probability of appearance $p(s_i)$ it is expedient to encode $-\log_2 p(s_i)$ with bats [2, p 17]. That is why the middle length of the code of element of block after application of any context-independent algorithm pursuant to the formula of Shannon [2, p. 611], can not be less *entropy of source*

$$H = -\sum_i p(s_i) \times \log_2 p(s_i). \tag{1}$$

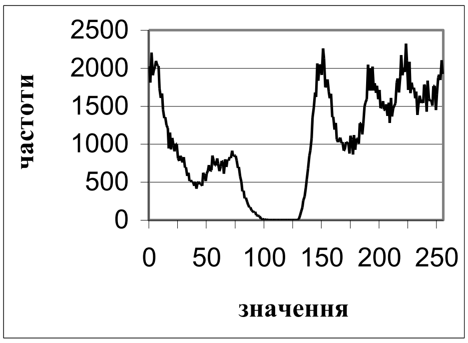
As it is known, the entropy of source diminishes with multiplying of the unevenness of division of probabilities (frequencies) between elements [1]. On our counts, application of context-independent| algorithm diminishes AC images on 33 % on the average.

To promote the efficiency of this code in the process of compression of images without losses it is tried by the use of *predictors*, which during a round forecast the value of brightness of each component of attendant pixels (for example, for the most widespread 24-bits images – it is brightness red, green and dark blue components, written down integers in separate bytes, using the value of brightness of those analised earlier components of contiguous pixels, as data of brightness has between itself the tallest degree of correlation [5, p. 675]. In the process of using of approach it is calculated and F_{ij} encoded the rejection Δ_{ij} of value of brightness of the next component of pixel from forecast select predictors value, that

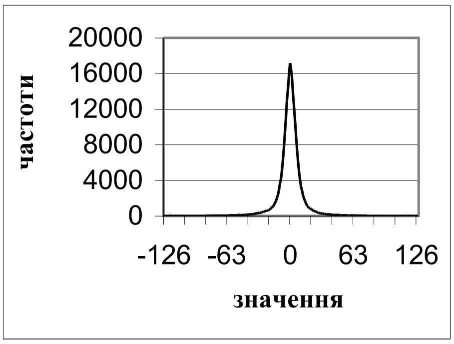
$$\Delta_{ij} = F_{ij} - predict_{ij} \tag{2}$$

(*i* and *j* run accordingly on all lines and columns of the components of pixels of the image). Contiguous pixels of images have similar colors more frequent, and near values of brightness of the proper components, that is why the value of prognosis often coincides with the value of brightness of duty component, more frequent – is near to this value and rarely – considerably differs from it (image 1). Most values Δ_{ij} appear near to the zero. In the same way the application of predictors multiplies the unevenness of division of probabilities of values of brightness and, as a result, diminishes the entropy more frequent (1).

Why do the values of brightness components of pixels deviate from values, forecast predictors? The point is that these rejections are more frequently conditioned by two objectively existent basic factors: by "strong" changes – trends and "weak" background vibrations – noise. "Two opposite types of models are possible that is why: payment of noise is insignificant in comparing with payment of evolution; payment of evolution is insignificant in comparing with payment of noise. In first case ...we will foresee a value ..., coming from ... the tendency which was folded, in the second – as equal to middle arithmetic ... previous elements" [3, p. 59].



à)



b)

Image 1. A division of frequencies of values of green components of image Lena.bmp:
a) before application of predictors (H=7,59 of bpb); b) after application of left-predictors (H=5,34 of bpb)

Except for it, during the successive round of pixels deviate can use the value of brightness from previous lines and on the left in a duty line (image 2) for prognostication which diminishes the efficiency of their application.

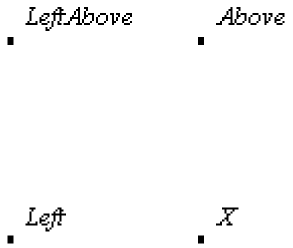


Image 2. Denotation of contiguous elements before the element X at the successive method of round of pixels of image

At such method of round of pixels, using the denotation of contiguous elements for an element *X* in accordance to image 2, it is more widespread for today predictors in the language of C are written down so:

```
typedef unsigned char ubyte;
ubyte LeftPredict(ubyte Left, ubyte Above, ubyte LeftAbove)
 {return Left;}
ubyte AbovePredict(ubyte Left, ubyte Above, ubyte LeftAbove)
 {return Above;}
ubyte AveragePredict(ubyte Left, ubyte Above, ubyte LeftAbove)
 {return (Left+Above)/2;}
ubyte PaethPredict(ubyte Left, ubyte Above, ubyte LeftAbove)
 {int pp=Left+Above-LeftAbove;
  int pa, pb, pc;
  pa=abs(pp-Left); pb=abs(pp-Above); pc=abs(pp-LeftAbove);
  if (pa<=pb && pa<=pc) return Left;
  else if (pb<=pc) return Above;
        else return LeftAbove;}
ubyte MedPredict(ubyte Left, ubyte Above, ubyte LeftAbove)
 {if (LeftAbove>=max(Left, Above)) return min(Left, Above);
  else if (LeftAbove<=min(Left, Above)) return max(Left, Above);
        else return Left+Above-LeftAbove; }
```

Predictor *LeftPredict* forecasts the value of duty element to be equal with a value on the left, predictor *AbovePredict* – to be equal with a value from above, predictor *AveragePredict* – to middle arithmetic these values. These three predictors belong to linear static predictors, they count expect middle arithmetic values of separate contiguous elements and that is why describe a noise model. Next two predictors belong to nonlinear static predictors, take into account tendencies in relation to a value, forecast in supposition of equalities of vibrations of backgrounds of diagonal elements, and, consequently, describe the mixed trends-noise model.

Predictor Pifa *PaethPredict* expects a value in a point *X*, coming from a plane which passes through the points *of Left*, *Above* and *LeftAbove* in three-dimensional space and forecasts one of these three values in direction of the least increase in relation to the expected value.

Predictor *MedPredict* tries to adapt oneself to the local horizontal and vertical ribs. The value *of Left* returns at the exposure of horizontal more frequent, and *Above* – at the exposure of vertical rib. If a rib is not found, the value of plane returns above a point *X*, which passes in three-dimensional space through the points *of Left*, *Above* and *LeftAbove*. First four from resulted predictors are used, for example, in the format of PNG, fourth – in archiving of WinRAR, the last – in the format of compression of JPEG-LS. Description of other predictors, which are used in the process of successive round of pixels, is possible to find in [1].

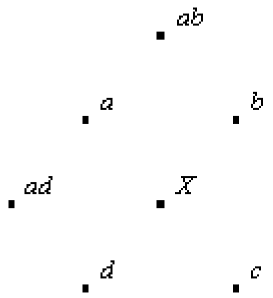
Progressing compression of images without losses. Symmetric hierarchical predictors

From one side the forward hierarchical compression of images enables to accelerate decoding, and from the other – to take into account the process of the use of the predictors value of the previous worked out elements from four, instead of only two different sides. For this reason for achieving a research's purpose we developed the effective chart of round of pixels and proper pixels, which will realize the making progress hierarchical compression of images without losses. In particular, for a making progress hierarchical round we offer a chart after which on the first layer of pixels of image worked out over consistently, beginning from the first, on lines from top to bottom, and in every line – successively from left to right with a step $h_1 = 2^k$, where k is determined from a condition $k = \left\lfloor \log_2\left(\frac{\max(\min(height;width);16)-1}{15}\right)\right\rfloor$, *height –an* amount of lines, *width –an* amount of columns of pixels of image. This step provides working on the first layer at least 16 pixels on each axe (as in icons), if an image has more considerable sizes. On the next layers ($l = \overline{2,k+1}$) intermediate pixels of images are processed in two passage-ways: on the first that is consistently worked over of them, which are contained on crossing of diagonals of squares with to[ps in contiguous pixels of previous layer with a step $h_l = 2^{k+2-l}$ both on lines and on columns, and on the second untilleds pixels treat between contiguous pixels of previous layer and first passage-way with the same step on columns and with twice diminished – on lines (image 3).

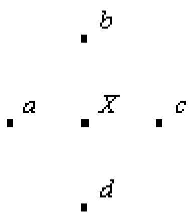
П	2	П	2	П	…
2	1	2	1	2	…
П	2	П	2	П	…
2	1	2	1	2	…
П	2	П	2	П	…
…	…	…	…	…	…

Image 3. A chart of the progressing hierarchical working of pixels of image on layers, beginning from the second:
П – pixels of the previous layer;
1 – pixels of the first passage-way of duty layer;
2 – pixels of the second passage-way of duty layer

The sequence of round of pixels of image, which is offered, enables not only to accelerate decoding, when sizes of area of conclusion are considerably less than sizes of images but also apply hierarchical predictors for prognostication of value of every element of attendant pixels (on image 4 it is marked through *X*) on all layers, beginning from the second. For description of these predictors we will designate the value of brightness of similar components of the **nearest** (contiguous) worked out before pixels from a previous and duty layer or passage-way by denotations *a*, *b*, *c*, *d*, *ab*, *ad* (image 4).



a)



b)

Rice 4. Charts of placing of the contiguous worked elements for an element X on layers, beginning from the second: a) for the first passage-way; b) for the second passage-way

By using these denotations, we will explore principles of prognostication of two basic symmetric hierarchical predictors, **oriented to base middle arithmetic** (noise constituent) of **those two opposite elements from the nearest four, which differ between themselves at least** (trendy constituent), so that more credible than all they belong to one object on an image. In the case when on the brightness of components of near pixels the weak vibrations of backgrounds have the greater influence, we will forecast the value of duty element by trendy-noise predictors *ProgresPredict1*, which returns base middle arithmetic. If the strong vibrations of brightness of pixels prevail, we will forecast the value of duty element by noise-trends predictors *ProgresPredict2*. Predictor determines and returns among four contiguous worked elements *a*, *b*, *c*, *d* the nearest value to base middle arithmetic, when such value is unique. If among the contiguous worked elements there are two nearest equidistant values to base middle arithmetic, then which repeats oneself more frequent returns among them, and when and the amount of their reiterations is identical – **less** from these two values. A general orientation on less values enables to displace un zero rejections Δ_{ij}, calculated accordant (2), in the side of positive values and to promote the unevenness of their division these same. By the language of C these predictors are written down so:

```
ubyte ProgresPredict1(ubyte a, ubyte b, ubyte c, ubyte d)
 {if (abs(a-c)<=abs(b-d))  return (a+c)/2
  else  return (b+d)/2; }

ubyte ProgresPredict2(ubyte a, ubyte b, ubyte c, ubyte d)
 {ubyte absac, absbd, maxac, minac, maxbd, minbd, prognozn;
  if (a<=c)   {absac=c-a; maxac=c; minac=a;}
  else   {absac=a-c; maxac=a; minac=c; }
  if (b<=d)   {absbd=d-b; maxbd=d; minbd=b; }
  else   {absbd=b-d; maxbd=b; minbd=d; }
  if (absac<=absbd)
   {if (minbd>=minac && minbd<=maxac)   return minbd;
    if (maxbd>=minac && maxbd<=maxac)   return maxbd;
    return minac; }
  else
   {if (minac>=minbd && minac<=maxbd)   return minac;
    if (maxac>=minbd && maxac<=maxbd)   return maxac;
    return minbd; }}
```

Let’s analyze the results of application of offered predictors of hierarchical round of pixels (table 1) for diminishing of the entropy of components of pixels of standard test set of Archive Comparison Test (ACT), which contains both synthesized (№№ 1 (with noises), 2, 7), and photorealistic (all other) images (for comparison in the line of *NonePredict* of this table the entropy of brightness components of pixels is resulted without application of predictors, and below there is the entropy of the same brightness after the use of most widespread predictors of successive round). Loading the TIFF-versions of these images is possible, for example, from http://www.compression.ca/act/act-files.html or from http://www.compression.ru/arctest/act/act-tif.htm.

Table 1

The entropy of brightness the components of pixels of images of the set of ACT after application of different predictors, bpb

Predictor	Layer	Passage-way	№ file								Middle entropy
			1	2	3	4	5	6	7	8	
NonePredict	1	1	7.54	4.65	7.75	7.50	7.66	7.32	5.99	7.66	7.01
LeftPredict	1	1	3.57	1.68	5.26	4.57	4.60	5.67	1.74	5.13	4.03
AbovePredict	1	1	3.68	2.23	4.92	4.60	4.72	5.97	1.86	4.94	4.12
AveragePredict	1	1	4.66	2.88	4.86	4.27	4.40	5.59	2.58	4.74	4.25
PaethPredict	1	1	1.90	1.51	4.90	4.26	4.29	5.50	1.35	4.61	3.54
MedPredict	1	1	1.90	1.54	4.84	4.15	4.20	5.43	1.37	4.50	3.49
ProgresPredict1	2	1	7.46	6.63	6.93	6.83	7.17	7.28	7.08	7.33	7.09
	2	2	7.05	5.80	6.19	6.43	6.07	6.96	6.11	7.10	6.46
	3	1	7.26	6.36	6.43	6.44	6.61	7.12	6.34	6.91	6.68
	3	2	6.70	5.66	5.80	6.02	5.69	6.81	5.27	6.53	6.06
	k+1	1	5.65	2.88	4.80	4.11	4.44	5.60	2.85	4.53	4.36
	k+1	2	3.99	1.89	4.45	3.61	3.83	4.93	1.73	3.87	3.54
	Together		4.89	2.56	4.73	4.04	4.22	5.41	2.47	4.42	4.09
ProgresPredict2	2	1	7.40	5.68	6.99	6.89	7.21	7.34	6.39	7.32	6.90
	2	2	7.04	5.18	6.18	6.49	6.12	6.98	5.22	7.02	6.28
	3	1	7.00	5.36	6.54	6.56	6.71	7.24	5.59	6.94	6.49
	3	2	6.69	4.97	5.88	6.11	5.82	6.88	4.27	6.52	5.89
	k+1	1	4.73	2.07	5.03	4.49	4.65	5.84	1.90	4.95	4.21
	k+1	2	0.23	1.19	4.68	4.03	4.11	5.24	0.96	4.39	3.10
	Together		2.75	1.81	4.94	4.41	4.50	5.66	1.64	4.83	3.82
ProgresPredict3	2	1	7.37	5.73	6.87	6.68	7.10	7.25	6.29	7.28	6.82
	3	1	7.08	5.44	6.27	6.46	6.50	7.13	5.28	6.90	6.38
	k+1	1	4.91	1.61	4.88	4.30	4.51	5.71	1.70	4.76	4.05
	Together		2.79	1.66	4.89	4.35	4.44	5.62	1.55	4.77	3.76
ProgresPredict4	2	1	7.46	5.80	7.02	6.86	7.10	7.41	6.15	7.44	6.91
	3	1	7.19	5.36	6.62	6.61	6.66	7.27	5.53	7.12	6.55
	k+1	1	2.55	1.82	5.17	4.71	4.78	6.03	1.44	5.22	3.97
	Together		2.22	1.74	4.99	4.48	4.54	5.73	1.50	4.91	3.76

As well as it followed to expect, the application of noise-trendy predictors is appeared to be more effective for synthesized and on some initial passage-ways of separate (№№ 3, 8) photorealistic images, as for them sharp overfills of brightness of the worked pixels on the scopes of the represented objects are characterized (influencing trends prevails), and trendy-noise – on the whole for photorealistic images, in fact their contiguous pixels have near, but different colors (influence of the noise prevails). The use of symmetric predictors substantially diminishes the entropy on the last layers in comparing to other known for today predictors of successive round, as these pixels take into account the influence of four equidistant pixels from different sides, instead of only pixels, on the left and from above, as in the case of successive round. Thus every worked out pixel of duty layer in the middle of image is used for treatment of eight pixels of next layer (instead of for maximum three next pixels, as for predictors of successive round).

The level of correlation of the brightness of the components of contiguous pixels on the last layers substantially is higher than the level of correlation of these brightness on the first layers, especially for the synthesized images, that is why the applications of predictors of successive round (and sometimes even waiver than their use) provide less entropy relatively progressing on initial layers, efficiency of the last with multiplying the number of layer grows although. That is why, on the whole not to multiply entropy, for the synthesized image № 2 it is expedient to give up application of predictors both on the second and on third layer, and for an image № 7 – on the first passage-way of the second layer.

Except for it, during implementation of the optimized variant of noise-trendy predictors *ProgresPredict2* on the average 6.33 operations of comparison are carried out, and trendy-noise predictors *ProgresPredict1* – only 3 such the operations. That the use trends-noise predictors in place of noise-trends diminishes the amount of such operations in over 2 times, that is why predictors *ProgresPredict1* it follows to give advantage before *ProgresPredict2* at the identical values of entropy after their application. For comparison: in the process of implementation nonlinear static predictors successive round *of PaethPredict* on the average executed 5.66, and *MedPredict* – only 5 operations of comparison. That application of trendy-noise predictors *ProgresPredict1* even instead of nonlinear predictors of successive round acceleration a code/decoding.

The use of asymmetric hierarchical predictors

To promote efficiency of noise-trends possibly due to an account on the first passage-way of every layer of values of the contiguous worked elements on the left and from above the same passage-way (on rice 4 and they are marked accordingly through *ad* and *ab*). In fact the colors of pixels are more frequent in all similar to the colors of near pixels exactly for horizontal lines or vertical lines and considerably rarer – to the colors of near pixels bias. This (instead of only by greater distances to the forecast element) is explain considerably lower efficiency of application of symmetric predictors on the first passage-way in relation to the second for every layer arbitrary image (see tabl. 1). To take into account the value of similar elements business and from below in relation to an element X it is impossible, as during the appendix of predictors to this element they are not yet worked (for this reason described farther predictors behave to asymmetric).

Will consider principles of prognostication two developed by us asymmetric noise-trends predictors. First from them, *ProgresPredict3*, the nearest value returns to base middle arithmetic not only among four nearest contiguous worked elements *and*, *b*, *c*, *d*, as *ProgresPredict2*, but also additionally among *ad* and *ab*, giving them advantage above other elements at identical distances. For the acceleration of calculations the least distance is to base middle arithmetic determined non-obvious, using circumstance that point of *z* the nearer to the middle between the points of *x* and *in* ($x \le y$), than less is a difference $|(y-z)-(z-x)| = |y-2z+x|$.

Second from asymmetric predictors, *ProgresPredict4*, determines the least after an absolute value diagonal increase in relation to the elements *of ad* and *ab* and a value in direction of this increase returns in relation to an element *X*, if an increase is given more small relatively the half of more small from the diagonal increases of the nearest elements ($\min(|a-c|,|b-d|)/2$). Thus among two identical diagonal increases advantage is given that is why, what is related to the less rectilinear increase round an element *X*. Otherwise predictor among four nearest contiguous worked elements *and*, *b*, *c*, *d* determines three the nearest to base middle arithmetic (more credible than all belong to one represented object) and returns a mean value among them. The language of C is give asymmetric hierarchical predictors written down so:

```
ubyte ProgresPredict3(ubyte a, ubyte b, ubyte c, ubyte d, ubyte ad, ubyte ab)
 {ubyte absac, absbd, maxac, minac, maxbd, minbd, prognozn, s, mins;
  if (a<=c)   {absac=c-a; maxac=c; minac=a; }
  else   {absac=a-c; maxac=a; minac=c; }
  if (b<=d)   {absbd=d-b; maxbd=d; minbd=b; }
  else   {absbd=b-d; maxbd=b; minbd=d; }
```

```
  if (absac<=absbd)
   {prognozn=ad; mins=abs(maxac+minac-2*ad);
    s=abs(maxac+minac-2*ab);
    if (s<mins)   {prognozn=ab; mins=s; }
    s=abs(maxac+minac-2*minbd);
    if (s<mins)   {prognozn=minbd; mins=s; }
    s=abs(maxac+minac-2*maxbd);
    if (s<mins)   {prognozn=maxbd; mins=s; }
    if (minac>=prognozn || prognozn>maxac)   return minac;
    return prognozn; }
  else
   {prognozn=ad; mins=abs(maxbd+minbd-2*ad);
    s=abs(maxbd+minbd-2*ab);
    if (s<mins)   {prognozn=ab; mins=s; }
    s=abs(maxbd+minbd-2*minac);
    if (s<mins)   {prognozn=minac; mins=s; }
    s=abs(maxbd+minbd-2*maxac);
    if (s<mins)   {prognozn=maxac; mins=s; }
    if (minbd>=prognozn || prognozn>maxbd)   return minbd;
    return prognozn; }}

ubyte ProgresPredict4(ubyte a, ubyte b, ubyte c, ubyte d, ubyte ad, ubyte ab)
 {ubyte absac, absbd, maxac, minac, maxbd, minbd, prognozn, minprD, minprGV, prD, prGV;
  if (a<=c)   {absac=c-a; maxac=c; minac=a; }
  else   {absac=a-c; maxac=a; minac=c; }
  if (b<=d)   {absbd=d-b; maxbd=d; minbd=b; }
  else   {absbd=b-d; maxbd=b; minbd=d; }
  minprD=abs(ab-a); minprGV=abs(a-d); prognozn=d;
  prD=abs(ab-b);
  if (prD<=minprD)
   {prGV=abs(b-c);
    if (prD<minprD || prGV<minprGV)   {minprD=prD; minprGV=prGV; prognozn=c; }}
  prD=abs(ad-a);
  if (prD<=minprD)
   {prGV=abs(a-b);
    if (prD<minprD || prGV<minprGV)   {minprD=prD; minprGV=prGV; prognozn=b; }}
  prD=abs(ad-d);
  if (prD<=minprD)
   {prGV=abs(d-c);
    if (prD<minprD || prGV<minprGV)   {minprD=prD; minprGV=prGV; prognozn=c; }}
  if (2*minprD<min(absac, absbd))   return prognozn;
  if (absac<=absbd)
   {if (abs(a+c-2*b)<=abs(a+c-2*d))   prognozn=b;
    else   prognozn=d;
    if (maxac<=prognozn)   return maxac;
    if (minac>=prognozn)   return minac;
    return prognozn; }
  else
   {if (abs(b+d-2*a)<=abs(b+d-2*c))   prognozn=a;
    else   prognozn=c;
    if (minbd>=prognozn)   return minbd;
```

```
if (maxbd<=prognozn)   return maxbd;
return prognozn; }}
```

The results of the application of offered asymmetric hierarchical predictors for diminishing of the entropy of components of pixels of set of images ACT on the first passage-way of separate layers are resulted in lower part of table 1 (we will remind, that on the second passage-way is used, as well as before, noise-trendy predictor *ProgresPredict2*, which takes into account the horizontal and vertical vibrations of brightness of the worked contiguous elements). From information of this table we can see that application on the first passage-way of every layer of asymmetric predictors in the place of symmetric enables to decrease entropy of the synthesized images on 0.24 bpb mainly due to the first passage-way of the last layer, where entropy for such images diminished on 0.98 bpb. But it is clear, that *ProgresPredict3* and *ProgresPredict4 are* used for prognostication of value of six, instead of four worked elements and execute accordingly on the average each time 11.33 and 18.83, instead of 6.33, operations of comparison, as *ProgresPredict2* and that is why accounts are settled longer. Except for it, predictor *ProgresPredict3 has* appeared more effective for the synthesized image ¹ 2, and *ProgresPredict4* – for images №№ 1, 7. That for the different synthesized images, and even for their different fragments and passage-ways, different asymmetric hierarchical predictors can appear more effective.

Complex application of hierarchical predictors

Research results say, that predictors should be used only from a passage-way and layer, when their application begins to diminish entropy, and among predictors it is necessary to elect such which provides the least entropy after the application. Moreover, the level of influencing and character of noises and trends can differ for the different fragments of image, that is why different hierarchical predictors can appear effective for them. We should underline that renouncing application of predictors follows for a whole passage-way, instead of for its separate fragments, as a value of brightness components of images without and after the use of predictors have different character of unevenness' division (see image 1). That is why in the process of progressing hierarchical compression it is expedient at first to set for every passage-way of duty layer, whether application of predictors diminishes entropy of brightness components of its pixels on the whole, whereupon, in the case of decision-making about application of predictors, to choose for every homogeneous fragment of pixels of passage-way exactly such hierarchical predictor, which maximally diminishes its entropy.

For a decision-making about application of predictors on every passage-way of duty layer we used symmetric hierarchical predictors, as they settle accounts quickly, and in the case of diminishing of the entropy elected by them **for every line of this passage-way** most effective among symmetric predictors and additionally for the first passage-ways – most effective between resulted all hierarchical predictors. The results of application of these methods of combining of hierarchical predictors for diminishing of entropy of components of pixels of set of representing ACT are resulted in table 2-4. For comparison at the beginning of these tables the results of similar appendix are resulted to the same images of combinations of successive predictors, which for every separate line provide the least entropy.

Table 2

The entropy of brightness components of pixels of representing the set of ACT after application of different combinations of predictors, bpb

Combination	№ file								Middle entropy
	1	2	3	4	5	6	7	8	
Successive predictors	1.89	1.46	4.80	4.15	4.20	5.43	1.33	4.50	3.47
Making progress symmetric predictors	2.75	1.80	4.73	4.04	4.21	5.41	1.64	4.42	3.63
Making progress symmetric and asymmetric predictors	2.26	1.69	4.73	4.04	4.19	5.41	1.53	4.42	3.53

Table 3

Time of code of predictors of pixels of representing the set of ACT by different combinations, s

Combination	№ file								Middle time
	1	2	3	4	5	6	7	8	
Successive predictors	6.16	10.22	2.30	3.41	2.26	3.46	4.18	3.41	4.43
Making progress symmetric predictors	4.83	7.91	1.81	2.69	1.76	2.80	3.30	2.74	3.48
Making progress symmetric and asymmetric predictors	6.04	10.00	2.30	3.35	2.36	3.46	4.18	3.41	4.39

Table 4

Time of decoding of predictors of pixels of representing the set of ACT by different combinations, s

Combination	Amount of layers	№ file								Middle time
		1	2	3	4	5	6	7	8	
Successive predictors	4	0.44	1.22	0.35	0.53	0.34	0.44	0.42	0.44	0.52
	all	0.44	1.22	0.35	0.53	0.34	0.44	0.42	0.44	0.52
Making progress symmetric predictors	4	0.11	0.06	0.05	0.05	0.06	0.11	0.11	0.05	0.08
	all	2.09	3.46	0.77	1.10	0.83	1.10	1.43	1.10	1.49
Making progress symmetric and asymmetric predictors	4	0.17	0.05	0.05	0.06	0.05	0.11	0.11	0.11	0.09
	all	2.64	4.39	0.77	1.10	0.82	1.15	1.87	1.16	1.74

As information of these tables testify, application of combinations of all hierarchical predictors diminishes the entropy in relation to most effective from these separate predictors on the average on the set of ACT on 0.23 bpb mainly due to the synthesized images. Additional application of asymmetric predictors slows a code on 26 % and decoding on 17 %, but diminishes entropy on 0.1 bpb. That is why for the rapid hierarchical compression of images it is expedient to apply combining of symmetric predictors, and for providing of maximal compression – additionally on the first passage-ways of every layer of combining of symmetric and asymmetric predictors. Application of hierarchical predictors provides less entropy relatively successive from 0.27 bpb for the synthesized images, but achieves the best values on this index on 0.06 bpb for photorealistic pictures. Progressing hierarchical method of round of pixels enables to execute decoding considerably quick from a successive round, when the sizes of area of conclusion are considerably less than sizes of images (for example, for filling of the area of output of 128 x 128 pixels (4 layers) such decoding is executed on the average in 5.78 times quicker, see table 4), as time of the hierarchical decoding depends on a less among the sizes of area of conclusion and sizes of image, and successive – only from the sizes of image.

Conclusions and prospects of subsequent researches

1. In the new versions of graphic formats and new formats of compression of images without losses it is expedient to realize progressing hierarchical compression, as it is allowed to accelerate decoding substantially, when sizes of area of conclusion are less than relatively sizes of images.
2. Diminishing of sizes of compressed progressing hierarchical method of images is achieved mainly on the last layers, as pixels, which are used by predictors, have with forecast pixels on the average the greatest level of correlation in relation to other layers.
3. On the first passage-way of every layer in the process of hierarchical compression to promote efficiency of application of symmetric predictors, which take into account the brightness of the worked nearest

pixels bias, is possible due to the account of brightness of worked thereon passage-way of the nearest contiguous pixels for horizontal lines and vertical lines.

4. During a context-independent compression without losses for the homogeneous fragments of images it is expedient to elect such predictor from a few alternative, which enables maximally to decrease entropy.

In the future, with the purpose of the additional diminishing of sizes of files of the compressed images without losses and acceleration of decoding, it is planned to adapt the context-independent methods of compression [3] to the hierarchical round of pixels of images and promote the efficiency of application of symmetric and asymmetric predictors by differences of color models [7].

1. Bredikhin D. Y. Compression the graphs without the losses of quality [Electronic resource] / D. Y. Bredikhin. – 2004. – Mode to access: http://www.compression.ru/download/articles/i_lless/bredikhin_2004_lossless_image_compression_doc.rar. 2. Gonsales R. Digital processing of images / R. Gonsales, R. Vuds. – M.: Tekhnosfera, 2005. – 1072 p. 3. Methods of compression of data. device of archivings, compression of images and video / D. Vatolin, A. Ratushnyak, M. Smirnov, Yu. Yukin. – M.: DIALOG-MIFI, 2003. – 384 p. 4. Miano Dzh. Formats and algorithms of compression of images in operations: Studies. manual / Dzh. Miano. – M.: Triumph, 2003. – 336 p., silt. – (Series: Practice of programming). 5. Prett Ý. Digital processing of images: Trudged. with eng. / Ý. Prett. – M.: the World, 1982. – Book 2, 480 p., silt. 6. Selomon D. Compression of data, images and sound / D. Selomon. – M.: Tekhnosfera, 2006. – 368 p. – (Series: World of programming: digital signal processing). 7. Shport'ko A. The use of differences of colors models for compression RGB-images without losses / A. Shport'ko // Selection and treatment of information. – 2009. – № 31 (107). – P. 90-97.