

RAPORT

INSERTIE REVERSIBILA DE MARE CAPACITATE IN IMAGINI CU NIVELE DE GRI

ETAPA I: Algoritmi marcare reversibilă de mare capacitate

Eforturile echipei de cercetare în această etapă s-au concentrat asupra dezvoltării algoritmilor de marcare reversibilă de mare capacitate. Menționăm ca realizări principale: elaborarea unui algoritm de inserție reversibilă de mare capacitate cu performanțe îmbunătățite față de algoritmi cunoscuți, dezvoltarea de algoritmi pentru etapa a doua de inserție, dezvoltarea unui algoritm care conservă histograma laplaciana a erorii de predicție. Algoritmii dezvoltați au fost testați pe seturi extinse de imagini. Obiectivele proiectului au fost îndeplinite.

Cu rezultatele obținute în această etapă avem o îmbunătățire de 0.25 bpp în medie față de rezultatele avute la demararea proiectului, respectiv 3.37 bpp față de 3.12 bpp pentru setul Kodak. Spațiul liber oferit de cele două nivele de inserție reversibilă integrate depășește compresoarele TIF, JPEG-LS și PNG și ne-am apropiat doar la 0.16 bpp, de JPEG 2000. De asemenea, pentru 4 din cele 24 de imagini Kodak (16,67%), algoritmul nostru depășește JPEG 2000.

Cercetările din această etapă au fost publicate într-un articol în revista *IEEE Trans. on Circuits and Systems II: Express Briefs*, revistă cotate ISI Q2, un articol prezentat la conferința IEEE EWDTS'2020, iar alte două articole sunt în fază avansată de redactare și urmează să fie trimise la reviste Q1.

Activitatea 1.1. Elaborare modul control praguri pentru inserție cu module multiple.

Din punctul de vedere al capacității de inserție, algoritmul cu module multiple de inserție pentru expandarea erorii de predicție, MM-PE, publicat de noi în SPIC 2019 [1], este cel mai performant algoritm publicat până acum și depășește semnificativ algoritmi existenți [2-4]. MM-PE a pornit de la ideea de a expanda eroarea de predicție la fiecare pixel până la limita maximă.

Expandarea va oferi capacitatea maximă de inserție dată de fiecare pixel din imagine, de $\log_2 n$ biti, unde n este:

$$n = \max\{m \mid 0 \leq x + (m - 1)p_e + w \leq 255\} \quad 1$$

Modulul de expandare n al fiecărui pixel trebuie cunoscut la detecție. Soluția noastră a fost de a construi o hartă de module, care compresată fără pierderi este inserată în imagine ca informație adițională. Pentru a reduce dimensiunea hărții, modulele au fost cuantizate optimizând diferența dintre spațiul oferit prin expandare și dimensiunea hărții compresate. MM-PE furnizează pentru imagini naturale cu nivele de gri în jur de 3 bpp.

Soluția memorării modului de expandare fiind costisitoare, am încercat estimarea modului de expandare. Dacă la inserție, modulul maxim se poate calcula direct, problema apare la detecție, unde valoarea pixelului nu este cunoscută. Soluția noastră este de a determina modulul nu pe baza valorii reale a pixelului, ci pe baza predicției. Pentru a putea estima modulul de expandare maxim am reintrodus pragul de control pentru eroarea de predicție, T , prag eliminat la MM-PE. Cu pragul T , și cunoscând semnul erorii de predicție, putem estima modulul care nu produce depășire pentru pixelii cu eroarea de predicție mai mică sau egală cu T . Soluția este ingenioasă, nu necesită memorarea modului, ci doar o hartă pentru a identifica pixelii care nu pot fi inserați ($|e| > T$). Mai departe am considerat optimizarea soluției, împărțind pixelii pe categorii în funcție de netezimea contextului de predicție și determinând pragul optim pentru fiecare plajă de netezime.

Pe lângă eliminarea etapei de compresie a hărții cu impact direct asupra complexității matematice, algoritmul permite și controlul capacitate/distorsiune – un avantaj considerabil față de [1] care a fost gândit pentru a asigura doar capacitatea maximă de inserție. Din punctul de vedere al capacității de inserție, noul algoritm este net superior algoritmului [1].

Față de varianta îmbunătățită IMM-PE elaborată în cadrul proiectului SWAT și încă nepublicată, în această etapă au fost aduse alte îmbunătățiri. În primul rând, trebuie menționată reducerea complexității matematice a părții de optimizare care scurtează drastic timpul de calcul. De asemenea, au fost aduse îmbunătățiri la procedura de ajustare a modulelor de expandare.

Un articol asupra noului algoritm este în curs de redactare și urmează a fi trimis la *IEEE Trans. on Image Processing*.

Activitatea 1.2 : Testare inserție/detecție cu control de praguri pentru module multiple.

Algoritmul de inserție de mare capacitate pentru etapa întâi a fost implementat și testat pe seturi de imagini cu nivele de gri. Pentru predicție am utilizat EDP pe contextul GAP (predictor liniar local pe context și bloc de învățare anticauzal). Rezultatele obținute pe setul Kodak de 24 de imagini sunt prezentate în tabelul de mai jos. Pentru comparație prezentăm și rezultatele pentru MM-PE [1] și pentru varianta inițială IMM-PE

Imagine	Versiuni anterioare [bpp]		V. nouă [bpp]
	MM-PE	IMM-PE	IMM-PE
1	2,13	2,20	2,29
2	3,37	3,47	3,59
3	4,01	4,24	4,32
4	3,29	3,40	3,51
5	2,19	2,36	2,44
6	2,36	2,80	2,95
7	3,85	3,90	4,05
8	1,87	2,08	2,11
9	3,55	3,75	3,80
10	3,52	3,71	3,77
11	2,76	2,91	3,07
12	3,18	3,68	3,76
13	1,38	1,56	1,60
14	2,41	2,58	2,58
15	2,59	3,45	3,45
16	3,25	3,38	3,38
17	2,94	3,33	3,33
18	2,17	2,41	2,65
19	2,95	3,12	3,17
20	1,16	3,79	3,97
21	2,88	2,99	3,08
22	2,92	3,01	3,11
23	3,90	4,18	4,27
24	2,54	2,73	2,94
Media bpp	2,79	3,12	3,22

Activitatea 1.3: Elaborare algoritm de inserție pentru imagini cu zgomot pentru nivelul doi de inserție.

Algoritmii de inserție eficienți se bazează pe expandarea erorii de predicție, respectiv expandarea de două ori pentru capacități de până la 1 bpp, expandare de n ori, cu $n > 2$, pentru algoritmii de capacitate mare. Expandarea erorii și adaugarea informației pot fi modelate ca zgomot aditiv. Pentru capacități mari de inserție, zgomotul adăugat are amplitudine mare. Pentru imagini cu nivel mare de zgomot, predictorii dezvoltati până acum dau erori mari și, în consecință, un al doilea nivel de inserție aduce câștiguri neglijabile.

Intrucât expandarea erorilor mari de predicție conduce la depășiri, propunem o schemă originală de inserție care să funcționeze pe perechi de pixeli și, în loc să expandeze erorile de predicție, expandează adaptiv fie suma, fie diferența erorilor. Ideea de la care am pornit este că, dacă avem două numere întregi x , y și cele două numere sunt comparabile în valoare absolută în sensul ca $2\min(|x|,|y|) \geq \max(|x|,|y|)$, dacă numerele au același semn, $|x-y| \leq x$ și $|x-y| \leq y$, iar dacă au semne diferite $|x+y| \leq x$ și $|x+y| \leq y$. Dacă $2\min(|x|,|y|) < \max(|x|,|y|)$ avem $\min(|x|,|y|) < |x-y| < \max(|x|,|y|)$ dacă numerele au același semn, sau $\min(|x|,|y|) < |x+y| < \max(|x|,|y|)$ pentru semne diferite.

Pe acest principiu am elaborat diferite scheme de expandare. Problema principală este să asigurăm reversibilitatea, adică, la detecție să putem identifica dacă a fost expandată suma sau diferența. Am rezolvat această problemă expandând doar pixelul cu eroarea de predicție cea mai mare. Astfel eroarea de predicție crește și se păstrează semnul erorii. Din informația de semn pentru erorile perechii de pixeli avem informația dacă s-a expandat suma sau s-a expandat diferența. Eroarea de predicție cea mai mare indică pixelul care a fost expandat. În acest mod pot fi expandate perechi disjuncte - (x_i, x_{i+1}) , (x_{i+2}, x_{i+3}) - sau nu - (x_i, x_{i+1}) , (x_{i+1}, x_{i+2}) .

Pe lângă inserția în perechi, a fost dezvoltat și un algoritm de inserție pe blocuri disjuncte cu lungime de bloc variabilă. Algoritmul pe blocuri mai aduce și alte inovații, de exemplu, ordonarea erorilor de predicție și expandarea după suma sau diferența față de eroarea cea mai mică în modul. Etapa de inserție se realizează în următorul mod:

1. Imaginea se parcurge linie cu linie de sus în jos, exceptând liniile pentru care nu se poate forma contextul de predicție.
2. Se alege lungimea blocului de pixeli ce va fi folosită pentru întreaga imagine.

3. Se parcurg pixelii din linia curentă pe coloane, exceptându-i pe cei pentru care nu se poate forma contextul de predicție sau ultimele coloane pentru care nu se poate completa un bloc de pixeli.
4. Pentru a completa un bloc de pixeli, se alege un grup mai mare, de exemplu de 3 ori lungimea blocului, iar pixelii se iau din trei în trei pentru a nu influența predicția vecinilor.
5. Grupa de pixeli va fi parcursă de trei ori pentru a forma trei blocuri de pixeli tratate succesiv.
6. Se calculează eroarea de predicție pentru toți pixelii din bloc, iar apoi se ordonează crescător în funcție de modulul erorii.
7. Pentru a putea diminua erorile pixelilor din bloc, se alege primul pixel nenul ca referință, care pentru imaginile cu o predicție slabă corespunde pixelului cu eroarea cea mai mică în majoritatea blocurilor.
8. Pixelii cu eroarea de predicție 0 și pixelul referință vor rămâne nemodificați.
9. Pentru ceilalți pixeli se calculează suma sau diferența dintre eroarea calculată și eroarea pixelului de referință. Dacă eroarea pixelului curent din bloc este de același semn cu referința, aceasta se scade, iar dacă cele două erori sunt de semn contrar se adună.
10. După calcularea erorii diminuate se va realiza expandarea erorii de predicție folosind ecuațiile obișnuite, cu adăugarea unui mic artificiu, acela de a scădea bitul pentru erorile negative. Pentru erorile de predicție diminuate pozitive (p'_x) din bloc, ecuațiile de inserție sunt:

$$\begin{cases} X = x + p'_x + b, & 0 < p'_x < t \\ X = x + t, & p'_x \geq t \end{cases}$$

Iar pentru erorile de predicție negative, ecuațiile de inserție sunt:

$$\begin{cases} X = x - t, & p'_x \leq -t \\ X = x + p'_x - b, & -t < p'_x < 0 \end{cases}$$

Etapa de decodare se realizează în următorul mod:

1. Imaginea se parcurge în sens invers, începând de la ultima linie în care s-a inserat informația.

2. Se aleg blocurile de pixeli similar etapei de inserție, dar grupele de pixeli se iau de la sfârșit spre început, pentru ca ultimul bloc marcat de pe linia respectivă să fie primul decodat.
3. După completarea pixelilor în bloc se calculează eroarea fiecăruia, iar apoi sunt sortați în funcție de modulul erorii.
4. Se caută pixelul referință, adică cel mai mic pixel cu eroare nenulă din bloc, iar toți pixelii cu eroarea nulă și referința vor rămâne la fel, deoarece nu au fost modificați în etapa de inserție.
5. Pentru ceilalți pixeli din bloc se calculează noua eroare de predicție, diminuată prin raportare la referință, iar apoi se decodează. Dacă au fost deplasați cu t nivele, se vor reface prin deplasarea în sens contrar, iar dacă s-a inserat în ei se extrage bitul și se reface valoarea originală.
6. Ecuațiile pentru refacerea pixelilor, p'_X pozitive sunt:

$$\begin{cases} b = p'_X \bmod 2, & 0 < p'_X < 2t \\ p_0 = \frac{p'_X - b}{2}, & 0 < p'_X < 2t \\ x = x - p_0 - b, & 0 < p'_X < 2t \\ x = X - t, & p'_X \geq t \end{cases}$$

iar pentru erorile de predicție negative, avem:

$$\begin{cases} x = X + t, & p'_X \leq -2t \\ b = p'_X \bmod 2, & -2t < p'_X < 0 \\ p_0 = \frac{p'_X + b}{2}, & -2t < p'_X < 0 \\ x = X - p'_X + b, & -2t < p'_X < 0 \end{cases}$$

Intenționăm să testăm algoritmul și pentru imagini cu texturi fine, imagini la care predicția dă rezultate slabe. Cum ideea expandării sumei/diferenței este o idee originală, avem în vedere elaborarea unui articol care, în funcție și de rezultatele obținute pentru imagini cu texturi, să fie publicat într-o revistă sau prezentat la o conferință selectivă.

Activitatea 1.4: Testare nivelul doi de inserție.

Nivelul doi de inserție a fost testat cu algoritmul de inserție cu expandarea sumei/diferenței pe blocuri, cu patru predictorii. Pe lângă expandarea sumei/diferenței pe blocuri

(ESDB) am testat și o schemă de inserție cu expandarea erorii de predicție într-o singură etapă (PEE).

Au fost aleși 4 predictorii, doi predictorii necauzali, P1 și P2, și anume rombul [5] și extensia la fereastra 3x3 (context utilizat în [6]), și doi predictorii cauzali, P3 și P4, respectiv jumătatea cauzală a ferestrei 3x3 și extensia ei cu pixelii adiacenți pe verticală și orizontală. După știința noastră, P3 și P4 nu au mai fost folosiți până acum în inserția reversibilă. Am propus predictorul P3 pentru a avea în contextul de predicție doar pixeli modificați în prima etapă de inserție. Predictorul P4 extinde contextul de predicție de la 4 la 6 pixeli și pleacă de la ideea de a îmbunătăți predicția în zonele uniforme, mediind pe un context mai larg pentru a elimina zgomotul introdus de prima etapă de inserție.

Cei patru predictorii sunt descriși în tabelul de mai jos.

<p style="text-align: center;">P1.</p> <table border="1" style="margin: auto;"> <tr><td></td><td>N</td><td></td></tr> <tr><td>V</td><td>x</td><td>E</td></tr> <tr><td></td><td>S</td><td></td></tr> </table> $\hat{x} = \left\lfloor \frac{N + E + S + V}{4} + \frac{1}{2} \right\rfloor$		N		V	x	E		S		<p style="text-align: center;">P2.</p> <table border="1" style="margin: auto;"> <tr><td>NV</td><td>N</td><td>NE</td></tr> <tr><td>V</td><td>x</td><td>E</td></tr> <tr><td>SV</td><td>S</td><td>SE</td></tr> </table> $\hat{x} = \left\lfloor \frac{N + NE + E + SE + S + SV + V + NV}{8} + \frac{1}{2} \right\rfloor$	NV	N	NE	V	x	E	SV	S	SE							
	N																									
V	x	E																								
	S																									
NV	N	NE																								
V	x	E																								
SV	S	SE																								
<p style="text-align: center;">P3.</p> <table border="1" style="margin: auto;"> <tr><td></td><td></td><td></td></tr> <tr><td></td><td>x</td><td>E</td></tr> <tr><td>SV</td><td>S</td><td>SE</td></tr> </table> $\hat{x} = \left\lfloor \frac{E + SE + S + SV}{4} + \frac{1}{2} \right\rfloor$					x	E	SV	S	SE	<p style="text-align: center;">P4.</p> <table border="1" style="margin: auto;"> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td>x</td><td>E</td><td>EE</td></tr> <tr><td>SV</td><td>S</td><td>SE</td><td></td></tr> <tr><td></td><td>SS</td><td></td><td></td></tr> </table> $\hat{x} = \left\lfloor \frac{E + SE + S + SV + EE + SS}{6} + \frac{1}{2} \right\rfloor$						x	E	EE	SV	S	SE			SS		
	x	E																								
SV	S	SE																								
	x	E	EE																							
SV	S	SE																								
	SS																									

Rezultatele pentru etapa a doua de inserție cu expandarea erorii de predicție (PEE) într-o singură trecere și cu schema cu expandare a sumei/diferenței pentru patru predictorii, P1-P4, sunt prezentate în tabelul următor.

Imagine	PEE-P1	PEE-P2	PEE-P3	PEE-P4	SDB-P1	SDB-P2	SDB-P3	SDB-P4
Kodak 1	0.2944	0.3205	0.3219	0.3138	0.2514	0.269	0.2673	0.2640
Kodak 2	0.3127	0.3123	0.3258	0.3126	0.2587	0.2601	0.2687	0.2589
Kodak 3	0.2342	0.2566	0.2619	0.2509	0.2009	0.2261	0.2228	0.2184
Kodak 4	0.2122	0.2062	0.2206	0.1979	0.1806	0.1809	0.1838	0.1699
Kodak 5	0.262	0.2556	0.2687	0.2470	0.2188	0.2213	0.2221	0.2069
Kodak 6	0.0752	0.0641	0.091	0.0666	0.062	0.0638	0.0739	0.0629
Kodak 7	0.3327	0.3465	0.3496	0.3413	0.2805	0.2953	0.2914	0.2886
Kodak 8	0.1728	0.1791	0.1993	0.1814	0.1466	0.1582	0.1661	0.1602
Kodak 9	0.2326	0.2392	0.2617	0.2387	0.1997	0.2199	0.2221	0.2132
Kodak 10	0.259	0.269	0.2794	0.2675	0.2242	0.2347	0.2350	0.2283
Kodak 11	0.344	0.3618	0.3636	0.3529	0.2823	0.2969	0.2972	0.2891
Kodak 12	0.1366	0.1348	0.1631	0.1502	0.1049	0.1096	0.1236	0.1149
Kodak 13	0.3006	0.3181	0.3243	0.3161	0.2528	0.265	0.2658	0.2613
Kodak 14	0.3535	0.3783	0.3806	0.3708	0.295	0.3135	0.3120	0.3047
Kodak 15	0.3127	0.3362	0.3411	0.3330	0.2699	0.2894	0.2878	0.2849
Kodak 16	0.3304	0.3354	0.3422	0.3307	0.2702	0.2732	0.2755	0.2690
Kodak 17	0.2818	0.3155	0.3119	0.3069	0.2365	0.266	0.2584	0.2556
Kodak 18	0.2751	0.2988	0.3003	0.2942	0.2291	0.2502	0.2474	0.2464
Kodak 19	0.3545	0.3711	0.3766	0.3623	0.2994	0.3158	0.3094	0.3038
Kodak 20	0.2409	0.2318	0.2515	0.2181	0.1973	0.1983	0.2048	0.1827
Kodak 21	0.2835	0.2864	0.2927	0.2811	0.2301	0.235	0.2382	0.2295
Kodak 22	0.3234	0.3478	0.3459	0.3398	0.2675	0.286	0.2838	0.2809
Kodak 23	0.2241	0.2107	0.2248	0.2101	0.1863	0.1816	0.1870	0.1807
Kodak 24	0.3281	0.354	0.357	0.3513	0.2786	0.2999	0.2972	0.2967
Medie [bpp]	0.2699	0.2804	0.2898	0.2765	0.2260	0.2379	0.2392	0.2321

După cum se observă, cele mai buna rezultate se obțin pentru inserție cu expandarea erorii de predicție într-o singură trecere cu predictorul P3, respectiv PEE-P3.

Activitatea 1.5: Integrare și validare algoritmi/programe inserție reversibilă de mare capacitate pentru imagini cu nivele de gri.

Cele două nivele de inserție au fost integrate, testate și am analizat rezultatele testelor. Astfel, am comparat rezultatele obținute cu varianta integrată față de rezultatele obținute cu

nivelul doi de testare și varianta inițială IMM-PE dezvoltată în cadrul proiectului SWAT. Față de cele 8 configurații din secțiunea 1.4, am selectat doar cele 3 cu rezultatele cele mai bune, respectiv o singură parcurgere cu predictorii P1, P2, P3. Rezultatele obținute sunt prezentate în tabelul următor

Imagine	NIV-I+II			IMM-PE + NIV-II		
	P1	P2	P3	P1	P2	P3
Kodak 1	0.2259	0.2307	0.2217	0.2944	0.3205	0.3219
Kodak 2	0.2063	0.1929	0.1948	0.3127	0.3123	0.3258
Kodak 3	0.1707	0.1717	0.1696	0.2342	0.2566	0.2619
Kodak 4	0.1460	0.1296	0.1329	0.2122	0.2062	0.2206
Kodak 5	0.1822	0.1605	0.1639	0.262	0.2556	0.2687
Kodak 6	0.0093	-0.0142	0.0000	0.0752	0.0641	0.091
Kodak 7	0.2218	0.2134	0.2106	0.3327	0.3465	0.3496
Kodak 8	0.1046	0.0960	0.1020	0.1728	0.1791	0.1993
Kodak 9	0.0944	0.0808	0.0888	0.2326	0.2392	0.2617
Kodak 10	0.1881	0.1845	0.1797	0.259	0.269	0.2794
Kodak 11	0.2379	0.2310	0.2266	0.344	0.3618	0.3636
Kodak 12	0.0937	0.0778	0.0876	0.1366	0.1348	0.1631
Kodak 13	0.2254	0.2222	0.2176	0.3006	0.3181	0.3243
Kodak 14	0.2425	0.2418	0.2335	0.3535	0.3783	0.3806
Kodak 15	0.2141	0.2160	0.2080	0.3127	0.3362	0.3411
Kodak 16	0.2115	0.2000	0.1953	0.3304	0.3354	0.3422
Kodak 17	0.1846	0.1924	0.1818	0.2818	0.3155	0.3119
Kodak 18	0.2463	0.2502	0.2440	0.2751	0.2988	0.3003
Kodak 19	0.2446	0.2390	0.2367	0.3545	0.3711	0.3766
Kodak 20	0.1598	0.1345	0.1438	0.2409	0.2318	0.2515
Kodak 21	0.1957	0.1812	0.1786	0.2835	0.2864	0.2927
Kodak 22	0.2842	0.2887	0.2813	0.3234	0.3478	0.3459
Kodak 23	0.1381	0.1132	0.1163	0.2241	0.2107	0.2248
Kodak 24	0.2437	0.2478	0.2417	0.3281	0.354	0.357
Capacitate medie (bpp)	0.1863	0.1784	0.1774	0.2699	0.2804	0.2898

Se observă că după modificarea nivelului I, performanțele nivelului II se reduc cu aproximativ 0.1 bpp. Se mai observă și că predictorii P1 și P2 depășesc P3, iar diferențele dintre P1 și P2 sunt neglijabile. În consecință vom folosi doar P1 în nivelul II de inserție împreună cu algoritmul cu control al modulelor din secțiunile 1.1, 1.2.

Prezentăm în tabelul următor, pentru setul Kodak spațiul liber obținut prin compresie fără pierderi cu 4 compresoare standard și prin inserție reversibilă cu IMM-PE și cu cele două nivele de inserție.

Imagine	Spatiu dupa compresie [bpp]				Inserție reversibilă [bpp]	
	TIF	JPEG-LS	PNG	JPEG-2000	IMM-PE	NIV-I+II
1	1,34	1,99	2,50	2,56	2,20	2,51
2	2,63	3,34	3,69	3,83	3,47	3,79
3	2,58	3,92	4,00	4,45	4,24	4,49
4	1,69	3,08	3,50	3,82	3,40	3,66
5	0,94	1,97	2,39	2,70	2,36	2,62
6	2,04	2,82	3,15	3,33	2,80	2,96
7	2,19	3,61	3,85	4,25	3,90	4,27
8	0,61	1,60	2,49	2,48	2,08	2,21
9	2,37	3,34	3,78	4,00	3,75	3,89
10	1,99	3,29	3,71	3,91	3,71	3,26
11	2,17	2,91	3,28	3,45	2,91	3,31
12	2,34	3,68	3,85	4,09	3,68	3,85
13	0,91	1,53	1,89	1,89	1,56	1,82
14	1,11	2,55	2,79	2,96	2,58	2,82
15	1,87	3,20	3,62	4,06	3,45	3,66
16	2,53	3,42	3,69	3,83	3,38	3,59
17	1,85	3,16	3,53	3,80	3,33	3,51
18	1,23	2,17	2,63	2,84	2,41	2,89
19	1,72	2,59	3,32	3,47	3,12	3,41
20	3,62	3,88	4,48	4,71	3,79	4,13
21	2,13	2,86	3,25	3,38	2,99	3,29
22	1,57	2,64	3,18	3,38	3,01	3,39
23	1,77	3,70	4,06	4,48	4,18	4,41
24	1,60	2,40	2,98	3,21	2,73	3,18
Media	1,86	2,90	3,31	3,53	3,12	3,37

Se observă o îmbunătățire de 0.25 bpp în medie față de rezultatele avute la demararea proiectului, respectiv 3.37 bpp față de 3.12. Spațiul liber oferit de cele două nivele de inserție reversibilă integrate, depășește compresoarele TIF, JPEG-LS și PNG și ne-am apropiat doar la

0.16 bpp de JPEG 2000. De asemenea, pentru 4 din cele 24 de imagini, respectiv 16,67%, algoritmul nostru depășește JPEG 2000.

Activitatea 1.6: Diseminare. Cercetările originale privind convergența algoritmilor adaptivi au fost publicate în

1. S. Radhika, F. Albu and A. Chandrasekar, "Steady State Mean Square Analysis of Standard Maximum Versoria Criterion Based Adaptive Algorithm." *IEEE Transactions on Circuits and Systems II: Express Briefs* (2020), factor de impact 2,814, ISI Q2;
2. F. Albu, Set-membership Sparsity-Aware Proportionate Normalized Least Mean Square Algorithms for Active Noise Control, *IEEE East-West Design & Test Symposium (EWDTS)*, Varna, Bulgaria.

Alte două articole sunt în fază finală de redactare și anume:

3. D. Coltuc, I.-C. Dragoi "On Preserving Histogram Aspect in Prediction Error Expansion RDH", redactat în proporție de 95%, în curs de de submitie la *IEEE Signal Processing Letters*, revistă ISI Q1 (anexat la raport);
4. I.-C. Dragoi, D. Coltuc, H. Coanda, "Adaptive Block Selection for High Fidelity Reversible Data Hiding based on Pixel-Value-Ordering", redactat în proporție de 85-90%, în curs de submitie la *IEEE Trans. on Information, Forensics and Security*, revistă ISI Q1.

Bibliografie

- [1] I. Caciula, H.G Coanda and D. Coltuc, "Multiple Moduli Prediction Error Expansion Reversible Data Hiding", *Signal Process.: Image Comm.*, vol.71, pp. 120-127, 2019.
- [2] X. Li, B. Yang and T. Zeng, "Efficient Reversible Watermarking Based on Adaptive Prediction-Error Expansion and Pixel Selection", *IEEE Trans. on Image Process.*, vol. 20, no. 12, pp. 3524–3533, 2011.

- [3] F. Peng, X. Li and B. Yang, "Adaptive Reversible Data Hiding Scheme Based on Integer transform", *Signal Process.*, vol. 92, no. 1, pp. 54-62, 2012.
- [4] X. Gui, X. Li and B. Yang, "A High Capacity Reversible Data Hiding Scheme Based on Generalized Prediction-Error Expansion and Adaptive Embedding", *Signal Process.*, vol. 98, pp. 370-380, 2014.
- [5] V. Sachnev, H. J. Kim, J. Nam, S. Suresh and Y. Q. Shi, "Reversible Watermarking Algorithm Using Sorting and Prediction", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, pp. 989-999, 2009.
- [6] B. Ou, X. Li, Y. Zhao, and R. Ni, "Reversible data hiding based on PDE, predictor," *J. Syst. Softw.*, vol. 86, no. 10, pp. 2700-2709, 2012.
- [7] I.-C. Dragoi and D. Coltuc, "Local-Prediction-Based Difference Expansion Reversible Watermarking", *IEEE Trans. on Image Processing*, vol. 23, no. 4, pp. 1779-1790, 2014.
- [8] X. Li, B. Yang and T. Zeng, "Efficient Reversible Watermarking Based on Adaptive Prediction-Error Expansion and Pixel Selection", *IEEE Trans. on Image Process.*, vol. 20, no. 12, pp. 3524-3533, 2011.