

KAYIPSIZ GÖRÜNTÜ SIKIŞTIRMA YÖNTEMLERİNİN KARŞILAŞTIRILMASI

Altan MESUT, Aydın CARUS

Trakya Üniversitesi, Mühendislik-Mimarlık Fakültesi, Bilgisayar Mühendisliği Bölümü
Edirne

e-posta: altanmesut@trakya.edu.tr

ÖZET

Farklı dosya tipleri ile yaptığımız sıkıştırma testlerinin sonucunda, tarafımızca geliştirilmiş kayıpsız bir veri sıkıştırma algoritması olan RSSDC algoritmasının, kendisi gibi sözlük tabanlı bir yaklaşım kullanarak sıkıştırma yapan LZW algoritmasına göre daha iyi sıkıştırma oranları elde ettiği görülmüştür. Bu çalışmada, hem RSSDC algoritmasının hem de kayıpsız sıkıştırma için kullanılan diğer yöntemlerin, kayıpsız görüntü sıkıştırma için performansları test edilmiş ve belirli bir tip görüntü için hangi yaklaşımın daha iyi sonuç verdiği ortaya konmuştur.

Anahtar Kelimeler: Kayıpsız Görüntü Sıkıştırma, JPEG, GIF, TIFF, LZW, RSSDC.

ABSTRACT

According to our compression tests with different file types, the compression ratio of the RSSDC algorithm, which is the dictionary-based data compression algorithm developed by us, is better than the compression ratio of the LZW algorithm, which is the most commonly used dictionary-based data compression algorithm. In this study, the performance of both RSSDC algorithm and the other methods that are used for lossless compression are tested for lossless image compression, and it is shown that which approach is more suitable for a particular type of image.

Key Words: Lossless Image Compression, JPEG, GIF, TIFF, LZW, RSSDC.

1. GİRİŞ

Görüntü sıkıştırmadan bahsedilirken ilk akla gelen, kayıplı görüntü sıkıştırma bir standart haline gelmiş olan JPEG yöntemidir. Bu yöntem fotoğraf gibi görüntülerin sıkıştırılması için çok elverişlidir. Fakat görüntü, sadece fotoğrafları kapsayan bir terim değildir. Şekiller gibi düşük renkli görüntüler söz konusu olduğunda, kayıplı sıkıştırmanın sonucunda ortaya çıkan görüntüdeki bozulma daha fazla gözle görünür hale gelmektedir. Bu yüzden bu tür görüntülerin sıkıştırılmasında genellikle kayıpsız yaklaşımlar tercih edilir.

Bu çalışmada, görüntüleri kayıpsız olarak sıkıştırmak için günümüzde en sık kullanılan yöntemleri, birbirinden farklı özelliklere sahip görüntü dosyalarından oluşan bir veri seti üzerinde test edip, hangi yöntemin hangi tip görüntü için daha kullanışlı olduğunu belirlemeyi amaçladık. Ayrıca, kendi geliştirdiğimiz RSSDC algoritmamızı da bu teste dahil ederek, popüler görüntü sıkıştırma teknikleri ile kıyaslandığında ne oranda başarılı olduğunu gözlemledik.

2. KARŞILAŞTIRILAN YÖNTEMLER

Karşılaştırmaya dâhil edilen yöntemler aşağıda verilmiştir:

- LZW tabanlı sıkıştırma kullanan GIF görüntü sıkıştırma formatı
- RLE, LZW, LZW+Predict, ZIP, JPEG sıkıştırma yöntemlerini kullanabilen TIFF görüntü sıkıştırma formatı
- ZLIB/Deflate tabanlı PNG görüntü sıkıştırma formatı
- Digram Kodlaması tabanlı RSSDC
- JPEG'in kayıpsız sıkıştırma yapan JPEG-LS versiyonu
- Kayıpsız JPEG2000 sıkıştırması
- WinZIP ve WinRAR sıkıştırma paketleri

Bu bölümde karşılaştırmada kullanılan yöntemler hakkında özet bilgiler yer almaktadır.

2.1. RLE (Run-Length Encoding)

Birbirini takip eden karakterlerin hepsini tutmak yerine, sadece karakteri ve kaç defa tekrar ettiğini tutma mantığına dayalı sıkıştırma yöntemidir. RLE algoritması yüksek sıkıştırma oranlarına ulaşmasa da, kolay uygulanabilir olması ve hızlı çalışması nedeniyle BMP, TIFF ve PCX gibi görüntü dosya formatlarının çoğu tarafından desteklenmektedir.

2.2. LZW (Lempel Ziv Welch)

Terry Welch 1984'te Unisys (o zamanki adı Sperry Corporation idi) için çalışırken, LZ78 [1] yaklaşımını yüksek performanslı disk ünitelerine uyarlamış ve ortaya çıkan yeni algoritma LZW [2] olarak kabul görmüştür. LZW hem sıkıştırma hem de açma performansı açısından LZ78 ailesinin en iyisi olmayı başarmıştır. Her tip veri üzerinde iyi sonuçlar veren bir algoritma olduğu için, kendisinden sonra gelen birçok algoritma LZW'yi temel almıştır. 1985 yılından beri Unisys LZW'nin patentini elinde bulundurmaktadır.

2.3. GIF (Graphics Interchange Format)

1987'de CompuServe tarafından geliştirilen GIF, LZW tabanlı çalışan algoritmalarından biridir. 1989'da 89a versiyonu yayınlanmıştır [3]. En fazla 8-bit renk derinliğine ($2^8 = 256$ renk) sahip görüntülere izin verir. Bu sebepten dolayı renkli fotoğraf görüntülerinin sıkıştırılmasında yetersiz kalsa da, birkaç rengin çoğunlukta olduğu grafiksel gösterim, basit şekiller, vb. gibi görüntülerin kayıpsız olarak sıkıştırılmasında kullanılmaktadır. Kullanım oranını düşüren bir diğer neden ise patent koruması altında olmasıdır.

2.4. TIFF (Tagged-Image File Format)

Aldus Corporation tarafından geliştirilmiştir. Aldus'un daha sonra Adobe Systems ile birleşmesi nedeniyle şu anda telif haklarına Adobe Systems şirketi sahiptir. 1988'de yayınlanan 5.0 sürümü ile RGB bitmap verilerini LZW algoritması kullanarak sıkıştırabilme yeteneğine kavuşmuştur. 1992'de yayınlanan 6.0 sürümü [4] ile de JPEG sıkıştırması eklenmiştir. TIFF-LZW'nin de GIF gibi patent sorunlarına yol açması sonucunda bazı girişimciler tarafından TIFF altında ZIP sıkıştırması geliştirilmiştir. Bu yapı da PNG gibi ZLIB/DEFLATE sıkıştırma teknikleri ile sıkıştırma yapar. TIFF 24-bit renk derinliğini desteklemesi sayesinde fotoğrafların kayıpsız olarak sıkıştırılmasında kullanılır.

2.5. PNG (Portable Network Graphics)

GIF ve altyapısındaki LZW algoritmalarının, patent anlaşmaları ile koruma altında olmaları, ücretsiz olarak kullanılabilen bir kayıpsız görüntü sıkıştırma standardı ihtiyacını gündeme getirmiştir. 1995 yılında Thomas Boutell, Scott Elliott, Mark Adler, Tom Lane, ve birçok başka girişimcinin başlattığı PNG projesi 1996'da sonuçlanmıştır. 1999'da yayınlanan 1.2 sürümünden [5] sonra ile birlikte kullanım oranı hızla artmıştır. 48-bit gerçek renk, 16-bit gri tonlama desteği ve yüksek oranda sıkıştırma becerisi sayesinde GIF ve TIFF standartlarından daha iyi olduğunu ispatlamıştır. Altyapısında ZLIB ve DEFLATE veri sıkıştırma teknikleri yer almaktadır.

2.6. RSSDC (Recursive Semi-Static Digram Coding)

2004 yılında tarafımızdan geliştirilmeye başlanan RSSDC [6], LZW gibi sözlük tabanlı bir sıkıştırma tekniğidir. Temelinde yine tarafımızdan geliştirilmiş olan ve statik sözlük kullanan Digram Kodlamasının [7] yarı-statik sözlüğe uyarlanmış hali olan SSDC (Semi-Static Digram Coding) yer alır. RSSDC, SSDC'nin yaptığı işi belirli bir sayıda tekrar ettiği için daha yavaş, fakat daha yüksek oranda sıkıştırır.

2.7. JPEG-LS (Lossless JPEG)

ISO/IEC tarafından geliştirilen, görüntüleri kayıpsız veya az kayıplı sıkıştırabilen JPEG-LS, 1998 yılının sonlarında tamamlanmıştır. Standardın temelini "Hewlett-Packard Laboratories" tarafından geliştirilen LOCO-I (LOW COMPLEXITY LOSSLESS COMPRESSION FOR IMAGES) [8] algoritması oluşturmaktadır. Kayıpsız JPEG2000'in ortaya çıkmasıyla yaygın bir kullanıma sahip olamamıştır.

2.8. Kayıpsız JPEG2000

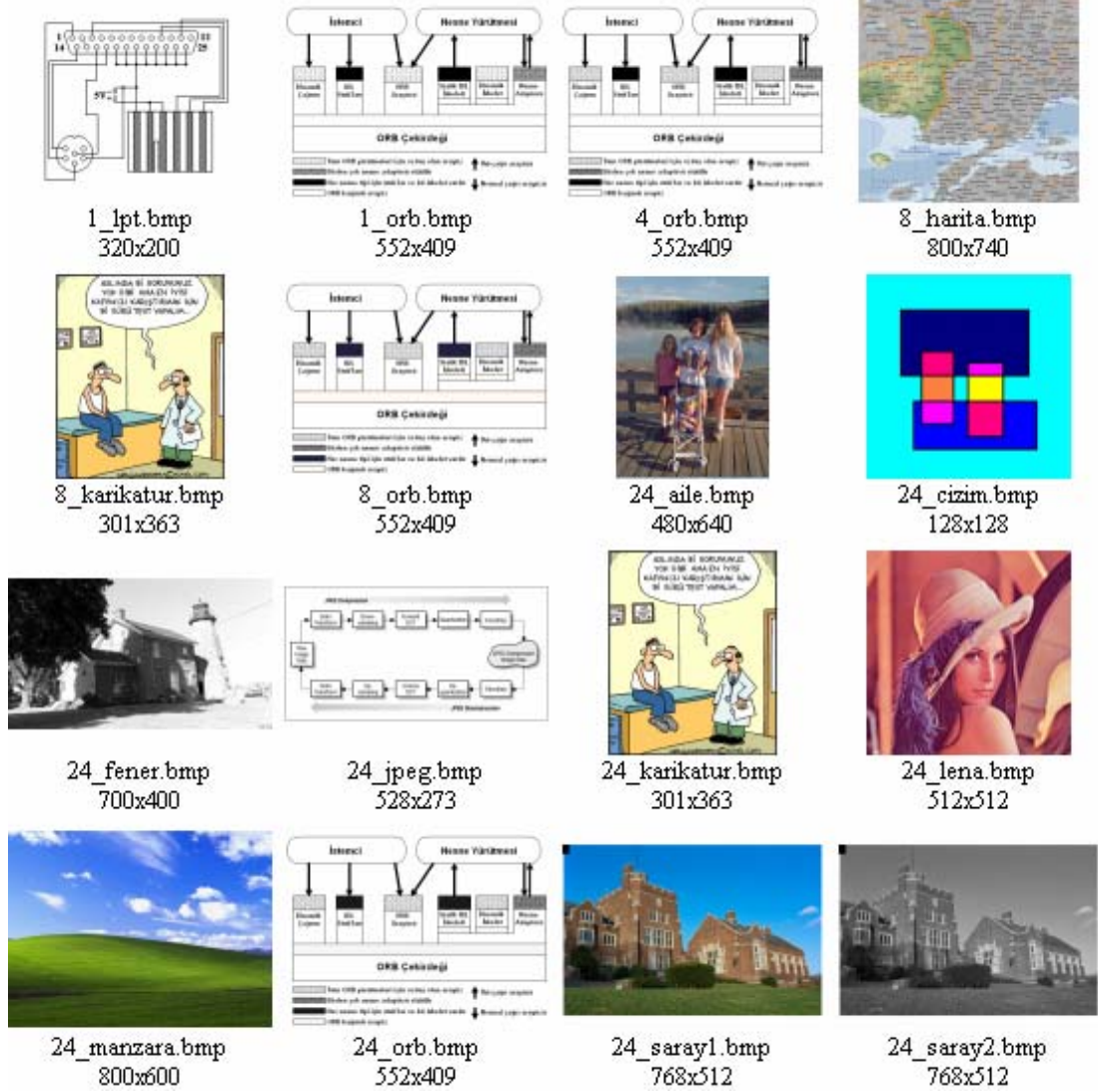
JPEG2000 [9] de JPEG gibi 'Joint Photographic Experts Group' adlı komite tarafından geliştirilmiştir. Wavelet teknolojisini temel alarak, bilinen en iyi sıkıştırma teknolojilerinin kullanılmasıyla oluşturulmuş bir kodlama sistemidir. JPEG standardının kısıtlamalarını gidermek ve düşük bit-oranlarında yüksek kalitede görüntüler elde etmek amacıyla tasarlanmıştır. Wavelet teknolojisinin avantajı sayesinde, DCT'de olduğu gibi sadece kayıplı sıkıştırma değil, kayıpsız sıkıştırma da yapabilmektedir. Ayrıca JPEG2000, istemci/sunucu görüntü uygulamaları için ve sınırlı kaynağa sahip olan kablosuz cihazlar için ilave fonksiyonel özelliklere sahiptir.

3. KARŞILAŞTIRMADA KULLANILAN GÖRÜNTÜ DOSYALARI

Karşılaştırmada sıkıştırılacak olan dosyalar Windows Bitmap (BMP) formatında seçilmiştir. Bunun sebebi hem bu formatın çok yaygın olarak kullanılması, hem de 1, 4, 8 ve 24 bit renk derinliklerini desteklemesidir.

Karşılaştırmada kullanılan görüntü dosyaları Şekil 1'de gösterilmiştir. Her görüntünün altında o görüntünün ait olduğu dosyanın ismi ve (genişlik) x (yükseklik) piksel değerleri yer almaktadır. Dosyaların isimlerinin başında yer alan numaralar, o dosyada 1 pikseli ifade etmek için kaç bit kullandığını (renk derinliğini) gösterir. Kullanılan 16 adet dosyadan iki tanesi 1-bit (1_lpt, 1_orb), bir tanesi 4-bit (4_orb), üç tanesi 8-bit (8_orb, 8_harita, 8_karikatur) ve geri kalan on dosya ise 24-bit renk derinliğindedir. Fakat 24-bit renk derinliğine sahip olan dosyaların hepsi fotoğraf değildir. 24_cizim, 24_orb, 24_karikatur ve 24_jpeg dosyaları, düz bir fon üzerine şekil ihtiva eden, az sayıda

renk içeren dosyalardır. Örneğin 24_cizim dosyası, kullanabileceği 16.7 milyon rengin sadece 8 tanesini kullanmaktadır. 24_fener ve 24_saray2 dosyaları ise gri-tonlamalı fotoğraf dosyalarıdır.



Şekil 1. Karşılaştırmada Kullanılan Görüntü Dosyaları

4. KARŞILAŞTIRMADA KULLANILAN PROGRAMLAR

Sonuçlar kısmında yer alan tablolarda verilen renk sayıları, o dosyanın renk derinliğini yada içerebileceği maksimum renk sayısını değil, o dosyada kullanılan ve birbirinden farklı olan tüm renklerin sayısını, yani tekil renk sayısını gösterir. Bu renk sayılarının bulunmasında ve PNG, TIFF ve GIF formatlarında sıkıştırma yapılmasında XnView v1.80.1 programından yararlanılmıştır.

Kayıpsız JPEG2000 sıkıştırmasında hem XnView hem de Jasper programları kullanılmış ve birbirine yakın sonuçlar ürettikleri görülmüştür. JPEG-LS/LOCO-I sıkıştırmasında ise Hewlett-Packard tarafından geliştirilen LOCO Encoder "locoe.exe" kullanılmıştır.

ZIP ve RAR sıkıştırmaları için WinRAR 3.30, RSSDC için ise tarafımızdan geliştirilen sıkıştırma programı kullanılmıştır.

5. SONUÇLAR

Karşılaştırmada sıkıştırma zamanları göz önüne alınmamış, sadece sıkıştırma oranlarına bakılmıştır. Test yapılırken RSSDC, JPEG2000 ve WinRAR yöntemlerinin diğerlerine göre daha yavaş sıkıştırma yaptıkları gözlenmiştir.

Karşılaştırma için 16 farklı görüntü dosyası, renk sayıları ve renk derinliklerine göre 5 farklı gruba ayrılmıştır. Belirli özelliklere sahip görüntü dosyalarının, belirli yöntemlerle daha iyi sıkıştırıldığını daha rahat gösterebilmek için böyle bir gruplamaya ihtiyaç duyulmuştur.

İlk grupta 1-bit ve 4-bit renk derinliğine sahip olan 3 adet dosya yer almaktadır. Bu 3 dosyanın karşılaştırmaya dâhil edilen tüm yöntemler ile sıkıştırılması sonucunda oluşan sıkıştırılmış boyutlar, byte cinsinden Tablo 1’de verilmiştir.

Tablo 1. 1-bit ve 4-bit renk derinliğindeki dosyaların sıkıştırma sonuçları

Dosya Adı	1_lpt	1_orb	4_orb	TOPLAM	
Tekil Renk Sayısı	2	2	4		
Dosya Boyutu (byte)	8.062	29.510	113.002	150.574	
Sıkıştırılmış Boyutlar (byte)	RAR	717	3.302	4.788	8.807
	ZIP	773	3.585	5.407	9.765
	PNG	815	3.636	5.614	10.065
	TIFF-ZIP	972	3.928	7.225	12.125
	RSSDC	1.125	4.873	6.587	12.585
	TIFF-LZW	1.916	6.340	11.085	19.341
	TIFF-LZWP	1.916	6.340	11.085	19.341
	GIF	3.016	7.676	9.890	20.582
	TIFF-RLE	6.982	15.386	28.614	50.982
	JPEG2000	13.235	51.830	54.294	119.359
	TIFF-JPEG	8.242	28.495	113.354	150.091

Bu tabloda ve diğer tablolarda, sıkıştırma yerine şişirme yapan yöntemler altı çizili olarak gösterilmiştir. Her 3 dosya için ve bu dosyaların toplamı için, en iyi 3 sıkıştırma oranı koyu griden açık griye doğru renklendirilerek ifade edilmiştir.

Tablo 1’de görüldüğü gibi, tüm dosyalarda ve toplamda en iyi sonucu RAR, ikinci en iyi sonucu ZIP ve üçüncü en iyi sonucu da PNG yöntemleri sağlamıştır. RSSDC, 1-bit renk derinliğinde TIFF-ZIP’in gerisinde kalmış, 4-bit renk derinliğinde ise daha başarılı olmuştur. Bu renk derinliklerinde TIFF-LZW ve TIFF-LZWP’nin aynı sonuçları verdiği görülmüştür. 8-bit tabanlı çalışan GIF yöntemi ve 24-bit tabanlı çalışan JPEG yöntemleri bu renk derinliklerinde başarısız sonuçlar üretmişlerdir.

İkinci grupta 8-bit renk derinliğine sahip dosyalar yer almaktadır. Bu gruptaki dosyalar için yapılan sıkıştırma sonuçları Tablo 2’de verilmiştir.

Tablo 2. 8-bit renk derinliğindeki dosyaların sonuçları

Dosya Adı	8_orb	8_karikatur	8_harita	TOPLAM	
Tekil Renk Sayısı	10	38	49		
Dosya Boyutu (byte)	226.846	111.430	593.078	931.354	
Sıkıştırılmış Boyutlar (byte)	RAR	5.783	14.716	61.250	81.749
	ZIP	7.202	17.631	72.404	97.237
	PNG	7.431	18.250	71.710	97.391
	TIFF-ZIP	10.910	19.726	84.337	114.973
	RSSDC	8.236	19.954	91.761	119.951
	GIF	12.564	20.252	88.448	121.264
	TIFF-LZW	17.322	21.173	102.028	140.523
	TIFF-LZWP	19.364	25.402	117.023	161.789
	TIFF-RLE	39.790	41.445	194.039	275.274
	JPEG2000	57.589	99.044	704.757	861.390
TIFF-JPEG	227.798	111.053	594.382	933.233	

Tablo 1’de olduğu gibi Tablo 2’de de ilk üç sıra yine RAR, ZIP, PNG şeklindedir. Aralarındaki fark PNG’nin 8_harita dosyasında ikinci en iyi sıkıştırma oranına sahip olmasıdır. Renk sayısı ve görüntünün karmaşıklığı arttıkça PNG’nin ZIP’e göre daha başarılı olduğu diğer tablolarda da görülecektir. İki tablo arasındaki bir diğer fark ise LZW-tabanlı sıkıştırma yapan GIF yönteminin, LZW ve LZWP ile desteklenen TIFF’in önüne geçmiş olmasıdır.

JPEG yöntemleri temelde sadece 24-bit renk derinliğini desteklemektedirler. Bu yöntemler, daha düşük renk derinliklerine sahip görüntüleri sıkıştırırken, önce bu görüntüleri 24-bit renk derinliğine göre tekrar kodlayıp daha sonra sıkıştırma işlemini uyguladıkları için, düşük renk derinliklerinde çoğu zaman sıkıştırma yerine şişirme yaptıkları Tablo 1 ve Tablo 2’de görülmektedir. JPEG-LS yöntemi ile sıkıştırma yapmak için kullandığımız LOCO Encoder, 24-bit haricindeki renk derinliklerinde sıkıştırma yapmadığı için JPEG-LS Tablo 1 ve Tablo 2’de yer almamıştır.

Üçüncü grup 24-bit renk derinliğinde olan, fakat az sayıda tekil renge sahip olan dosyalardan meydana gelmektedir. Bu gruptaki dosyalar için yapılan karşılaştırmanın sonuçları Tablo 3’te verilmiştir.

Bu grup için de en iyi sıkıştırma yöntemi RAR olmuş, fakat önceki tablolardan farklı olarak ZIP ikinciliği RSSDC’ye kaptırmıştır. Önceki grupta TIFF-LZW ve TIFF-LZWP’yi geçmiş olan GIF, bu grupta TIF-ZIP’i de geçmiştir. 24-bit renk derinliğinde daha iyi sıkıştırma oranları sağlamaya başlayan JPEG ailesi, buna rağmen düşük tekil renk sayılarına sahip olan bu grupta da diğer yöntemlere göre geride kalmıştır.

Tablo 3. 24-bit renk derinliğinde ve az sayıda tekil renge sahip olan dosyaların sonuçları

Dosya Adı	24_cizim	24_orb	24_karikatur	TOPLAM	
Tekil Renk Sayısı	8	22	38		
Dosya Boyutu (byte)	49.206	677.358	328.206	1.054.770	
Sıkıştırılmış Boyutlar (byte)	RAR	201	8.818	19.492	28.511
	RSSDC	372	8.477	21.616	30.465
	ZIP	301	10.596	21.771	32.668
	GIF	1.673	13.237	20.252	35.162
	PNG	445	12.538	27.655	40.638
	TIFF-ZIP	781	19.392	23.811	43.984
	TIFF-LZW	3.124	34.710	29.890	67.724
	TIFF-LZWP	2.200	42.429	31.911	76.540
	JPEG2000	4.926	56.462	99.044	160.432
	JPEG-LS	1.166	75.657	97.587	174.410
	TIFF-RLE	49.191	48.390	252.874	350.455
	TIFF-JPEG	8.392	315.210	246.530	570.132

Arka planı beyaz olan 24_orb dosyasında 1/14 gibi bir sıkıştırma oranı sağlayan TIFF-RLE, arka planı beyaz olmayan 24_cizim ve 24_karikatur dosyalarında çok düşük sıkıştırma oranları sağlayabilmiştir. Bunun nedeni 24-bit renk derinliğinde RGB formatının kullanılmasıdır. Beyaz renk RGB formatında 255,255,255 olarak saklandığı için, beyaz bir arka plan dosyada birbirini takip eden yüzlerce 255 karakteri anlamına gelmektedir. Bu da RLE için aranılan durumdur. Fakat 24_cizim dosyasındaki mavi arka plan 255,255,0 olarak dosyada saklanmaktadır. Bu yüzden seri bozulmakta ve sıkıştırma oranı çok düşük olmaktadır.

Tablo 4. 24-bit renk derinliğinde ve 200-256 arası tekil renge sahip olan dosyaların sonuçları

Dosya Adı	24_jpeg	24_fener	24_saray2	TOPLAM	
Tekil Renk Sayısı	201	235	255		
Dosya Boyutu (byte)	432.486	840.054	1.179.702	2.452.242	
Sıkıştırılmış Boyutlar (byte)	JPEG2000	28.689	151.081	220.449	400.219
	RAR	11.531	233.453	329.384	574.368
	GIF	16.697	228.137	334.758	579.592
	PNG	17.262	253.907	344.461	615.630
	ZIP	13.890	259.796	360.820	634.506
	RSSDC	17.082	270.644	408.143	695.869
	TIFF-ZIP	19.598	330.469	452.729	802.796
	TIFF-LZWP	32.983	320.514	470.641	824.138
	TIFF-LZW	34.169	374.172	505.145	913.486
	JPEG-LS	53.477	427.215	633.104	1.113.796
	TIFF-RLE	38.832	446.022	684.360	1.169.214
	TIFF-JPEG	153.590	545.628	786.432	1.485.650

Dördüncü grup, 24-bit renk derinliğinde ve 200-256 arası tekil renk sayısına sahip olan dosyalardan oluşmaktadır. Bu dosyalardan biri şekil türünde (24_jpeg) diğer ikisi ise gri-tonlamalı (diğer bir

deyişle siyah/beyaz) fotoğraf türündedir. Bu üç dosya için yapılan karşılaştırmanın sonuçları Tablo 4'te verilmiştir.

Tablo 4'te görüldüğü gibi, 24_jpeg dosyasında yedinci en iyi sıkıştırma oranına ulaşabilen JPEG2000, fotoğraf türündeki diğer iki dosyada ise açık farkla birinci olarak, toplamda da birinci olmuştur. Fotoğraf türündeki dosyalarda RAR ve GIF birbirlerine yakın sonuçlar elde etmişler, ikinci ve üçüncü sıraları paylaşmışlardır. Şekil türündeki 24_jpeg dosyası her ne kadar 201 farklı renge sahip olsa da, önceki gruplardaki görüntü dosyaları gibi az detaya sahip olması sebebiyle, sözlük tabanlı yöntemler tarafından daha iyi sıkıştırılmış olması şaşırtıcı değildir.

Beşinci ve son grupta renkli fotoğraf dosyaları yer almaktadır. Bu grupta yer alan dört dosya için yapılan karşılaştırmanın sonuçları Tablo 5'te verilmiştir. 8-bit renk derinliğini destekleyen GIF formatı, 256'dan fazla sayıda tekil renk içeren görüntülerde ancak kayıplı sıkıştırma yapabildiği için, bu grupta GIF ile yapılan sıkıştırma oranları dikkate alınmamıştır.

Tablo 5. 24-bit renk derinliğinde ve çok sayıda tekil renge sahip olan dosyaların sonuçları

Dosya Adı	24_manzara	24_saray1	24_aile	24_lena	TOPLAM	
Tekil Renk Sayısı	46.841	85.427	101.565	148.279		
Dosya Boyutu (byte)	1.440.054	1.179.702	921.654	786.486	4.327.896	
Sıkıştırılmış Boyutlar (byte)	JPEG2000	377.390	649.372	523.733	444.585	1.995.080
	JPEG-LS	511.338	620.991	528.733	445.799	2.106.861
	RAR	421.700	642.859	560.331	485.059	2.109.949
	PNG	496.153	682.892	565.194	476.216	2.220.455
	TIFF-JPEG	486.958	770.240	644.066	518.720	2.419.984
	TIFF-LZWP	598.830	897.161	761.487	652.358	2.909.836
	ZIP	750.367	847.724	764.478	733.627	3.096.196
	TIFF-ZIP	820.075	889.929	780.518	745.663	3.236.185
	TIFF-LZW	1.145.069	1.076.069	<u>981.005</u>	<u>953.414</u>	4.155.557
	RSSDC	1.389.214	1.065.647	<u>980.735</u>	<u>846.809</u>	4.282.405
TIFF-RLE	<u>1.455.623</u>	1.144.516	<u>934.179</u>	<u>797.941</u>	4.332.259	

Dördüncü grupta JPEG2000'nin gösterdiği başarıyı yakalayamayan JPEG_LS, beşinci grupta JPEG2000'nin ardından ikinci sırayı alarak, esasen renkli fotoğraflar için etkin olduğunu göstermiştir. Diğer dört tablonun hepsinde son sırada yer alan TIFF-JPEG, ancak bu tabloda üst sıralarda yer alabilmiştir. Yani, JPEG ailesinin üç üyesi ancak renkli fotoğraflar söz konusu olduğunda ilk beşin içine girebilmişlerdir. Tüm gruplarda ilk üç sırada yer almayı başaran RAR, bu grupta da başarılı olmuştur. İlk gruplarda başarılı olan sözlük tabanlı yöntemlerin (LZW ve RSSDC), bu grupta başarısız olduğu görülmektedir.

Tüm grupların sonuçları değerlendirildiğinde;

- Sıkıştırma paketleri içinde RAR, görüntü formatları içinde ise PNG, tüm gruplarda başarılı sonuçlar ortaya koymuş, her tip görüntü için kullanılabilir olduklarını ispatlamışlardır.
- ZIP genellikle RAR'ın gerisinde kalmış, grup numarası arttıkça RAR'ın daha da gerisine düşmüş, beşinci grupta aralarındaki fark %50'ye yaklaşmıştır.
- 20 seneyi aşkın bir süredir kullanılan GIF, renkli fotoğraflar haricinde halen kullanılabilir bir alternatif olduğunu göstermiştir.

- TIFF-LZW'nin farklı bir versiyonu olan ve ilk üç grupta onun gerisinde kalan TIFF-LZWP, renk sayısı arttıkça TIFF-LZW'ye oranla daha başarılı olmuştur.
- TIFF-ZIP, veri sıkıştırma versiyonu olan ZIP'in gerisinde ama ona yakın sonuçlar elde etmiş, fakat tüm dosyalar için PNG'nin gerisinde kalarak, görüntü formatları arasında bir alternatif olma şansını kaybetmiştir. Son grupta TIFF-LZWP'nin de gerisine düşmüştür.
- Basit bir sıkıştırma mantığı kullanan TIFF-RLE, tüm gruplarda son sıralarda yer almaktan kurtulamamıştır.
- RSSDC en büyük başarısını üçüncü grupta, yani yüksek renk derinliği ve düşük renk sayısına sahip olan dosyalarda göstermiştir.
- JPEG ailesinin kayıpsız sıkıştırma yapan versiyonları da, kayıplı versiyonları gibi fotoğraf görüntülerinde başarılı olduklarını göstermişlerdir. İçlerinde en iyisi olan JPEG2000'nin kayıpsız versiyonu, özellikle siyah/beyaz fotoğraflar için alternatifsizdir. JPEG-LS ve TIFF-JPEG ise sadece renkli fotoğraflarda etkilidirler.

Pratikte, fotoğraflar söz konusu olduğunda, kayıpsız sıkıştırmak yerine, dikkatli bakılmadıkça fark edilemeyecek oranlarda küçük kayıplarla sıkıştırma yaparak, kayıpsız JPEG versiyonlarına göre bile birkaç kat daha fazla sıkıştıran kayıplı JPEG versiyonlarını kullanmak daha uygundur. Bu açıdan bakıldığında, ilk üç grubun sonuçları daha anlamlı hale gelmektedir. Çünkü kayıpsız sıkıştırmanın en çok tercih edildiği durumlar, şekiller gibi az renk kullanan ve az detay içeren dosyalardır. Giriş kısmında da belirttiğimiz gibi, bu tür görüntülerde kayıplı sıkıştırmanın yarattığı görüntüdeki bozulma daha fazla gözle görünür hale gelmekte ve kayıpsız sıkıştırmayı gerekli kılmaktadır.

Tarafımızdan geliştirilen RSSDC algoritması aslında görüntü sıkıştırma için özel olarak tasarlanmış bir algoritma değildir. LZW gibi hem metin, hem de görüntü sıkıştırmada kullanılabilen genel bir sıkıştırma yöntemidir. RSSDC bir görüntü formatı haline getirilirse (Birçok görüntü formatında olduğu gibi, renk derinliği, yükseklik, genişlik gibi değerler dosyanın başlığında saklanırsa), kayıpsız görüntü sıkıştırması için LZW tabanlı sıkıştırma yapan GIF ve TIFF yöntemlerine iyi bir alternatif olabileceği yapılan testlerde görülmüştür.

6. KAYNAKLAR

- [1] J. Ziv, A. Lempel. Compression of Individual Sequences via Variable-Rate Coding. IEEE Transactions on Information Theory, IT-24(5):530-536, Eylül 1978.
- [2] T. A. Welch. A Technique for High-Performance Data Compression. IEEE Computer, 17(6):8-19, Haziran 1984.
- [3] Graphics Interchange Format(sm), Version 89a, CompuServe Incorporated, Columbus, Ohio, 31 Temmuz 1990.
- [4] TIFF Revision 6.0, Aldus Corporation, Seattle, 3 Haziran 1992.
- [5] PNG (Portable Network Graphics) Specification, Version 1.2, PNG Development Group, G. Randers-Pehrson, et. al., Temmuz 1999.
- [6] A. Mesut, A. Carus. A New Approach to Dictionary-Based Lossless Compression. UNITECH'04 International Scientific Conference, Gabrovo, Kasım 2004.
- [7] K. Sayood. Introduction to Data Compression. San Francisco, California, Morgan Kaufmann, 1996.
- [8] M. J. Weinberger, G. Seroussi, G. Sapiro. The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS. Computer Systems Laboratory, HPL-98-193, Kasım 1998.
- [9] M. W. Marcellin, M. J. Gormish, A. Bilgin, M. P. Boliek. An Overview of JPEG-2000. Proc. of IEEE Data Compression Conference, pp. 523-541, 2000.