Fırat Üniversitesi Müh. Bil. Dergisi
34(2), 687-697, 2022

Araştırma Makalesi
https://doi.org/10.35234/fumbd.1122197

Trivium Algoritması Kaynaklı Rastgele Permutasyon Üretimiyle Görüntü Şifreleme Uygulaması

Taha ETEM[1*], Turgay KAYA[2]
[1] Elektrik-Elektronik Mühendisliği, Muş Alparslan Üniversitesi, Mühendislik Fakültesi, Muş, Türkiye
[2] Elektrik-Elektronik Mühendisliği, Fırat Üniversitesi, Mühendislik Fakültesi, Elazığ, Türkiye
[*1] t.etem@alparslan.edu.tr, [2] tkaya@firat.edu.tr

(Geliş/Received: 27/05/2022; Kabul/Accepted: 10/08/2022)

Öz: Görüntü şifreleme sistemlerinin popülerliği gün geçtikçe katlanarak artmaktadır. Farklı algoritmalar ve sistemler yardımıyla yeni şifreleme sistemleri tasarlansa da literatürdeki ihtiyacın doldurulması mümkün olmamıştır. Görüntü iletiminin gün geçtikçe yaygınlaşması yeni şifreleme sistemlerine olan ihtiyacı arttırmaktadır. Ayrıca görüntü iletimindeki mahremiyet ihtiyacı diğer veri türlerinden daha fazladır. Yapılan bu çalışmada Trivium algoritması yardımıyla üretilen rastgele sayı dizileri öncelikle permutasyon oluşturularak piksel karıştırma işlemlerinde kullanılmıştır. Ayrıca üretilen bitler şifreleme işleminde kullanılmak için anahtar üretiminde kullanılmıştır. Üretilen bitlerin rastgelelik koşullarını sağlayıp sağlamadığı NIST testleri yardımıyla başarılı olarak sonuçlandırıldıktan sonra tasarlanan sistemin güvenlik analizleri farklı teknikler yardımıyla gerçekleştirilmiştir. Histogram analizi sonuçlarına göre sadece piksel karıştırma işlemiyle kriptolojik işlem yapan sistemlerin güvenlik açıkları gösterilmiştir. Tüm sistemin uygulanması durumunda başarılı bir görüntü şifreleme işleminin gerçekleştirilebileceği tespit edilmiştir

Anahtar kelimeler: Rastgele Sayı Üreteci, Trivium Algoritması, Görüntü Şifreleme, Piksel Karıştırma, Kriptanaliz.

Image Encryption Application with Random Permutation Generation Based on Trivium Algorithm

Abstract: The popularity of image encryption systems is increasing exponentially day by day. Although new encryption systems have been designed with the help of different algorithms and systems, it has not been possible to fill the need in the literature The widespread use of image transmission increases the need for new encryption systems. In addition, the need for privacy in image transmission is more than other data types. In this study, random number sequences produced with the help of the Trivium algorithm were first used in pixel mixing processes by creating permutations. In addition, the generated bits are used in key generation for use in encryption. Whether the generated bits meet the randomness conditions or not has been successfully concluded with the help of NIST tests, and the security analyzes of the designed system have been carried out with the help of different techniques. According to the results of the histogram analysis, the security vulnerabilities of the systems that perform cryptographic operations only with pixel scrambling are shown. It has been determined that a successful image encryption process can be performed if the entire system is implemented.

Key words: Random Number Generator, Trivium Algorithm, Image Encryption, Pixel Scrambling, Cryptanalysis.

1. Giriş

Şifreleme sistemleriyle ilgili günümüzde birçok çalışma yapılmaktadır. Belirli bir düzene dayalı şifreleme algoritmalarının geliştirilmesi de oldukça önemli bir parametre olarak öne çıkmaktadır. Bu amaçlarla rastgele sayı üreteçlerinin şifreleme algoritmaları içerisinde yer aldığı roller de giderek artmaktadır [1–8]. Şifreleme sistemlerinde güvenliği sağlamak için farklı parametreler mevcuttur. Güvenli bir şifreleme sisteminin tasarlandığından bahsedebilmek için belirli güvenlik testlerinden başarılı olma şartı aranmaktadır [9–14]. Bununla birlikte kaos tabanlı uygulamaların rastgele sayı üreteçlerinin tasarımında ve şifreleme algoritmalarının temelinde kullanılması bu alandaki çalışmaların yaygınlaşmasını sağlayarak önemini arttırmıştır [15–19].

Şifreleme sistemleri kullanılırken farklı yöntemlere başvurulabilir. En temel şifreleme sistem farklılığı anahtar kullanımına dayalı olandır. Şifreleme ve şifre çözme işlemleri için aynı anahtarın kullanıldığı durumlarda simetrik şifreleme yöntemi, farklı anahtarların kullanıldığı durumlarda is asimetrik şifreleme yöntemi kullanılmış olur. Şifreleme algoritmasının yapısı da bu tercih doğrultusunda temelden değişiklik gösterecektir.

* Sorumlu yazar: t.etem@alparslan.edu.tr. Yazarların ORCID Numarası: [1] 0000-0003-1419-5008, [2] 0000-0002-7732-6194

Literatürdeki şifreleme uygulamaları ise genellikle görüntü şifreleme sistemleri üzerine yoğunlaşmıştır. Görüntü şifreleme sistemlerinin yaygınlaşmasındaki en önemli faktör görüntü iletiminin yaygın olması ve bu veri türünde mahremiyete daha fazla ihtiyaç duyulması gelmektedir. Ayrıca görüntü şifreleme uygulamalarının çıktılarını akademik makalelerde incelemenin daha kolay olması da bu alandaki çalışmaları arttırmaktadır. Örneğin, karmaşık bir veri grubunun şifrelenmesi sonucunda ortaya çıkan çıktılar o veri türüne hakim olmayan bir araştırmacı için orijinal veriden ayırt edilemeyebilir. Ancak bir görüntü şifrelendiğinde bu görüntünün temel olarak başarılı bir şekilde şifrelenip şifrelenmediğini gözlemlemek daha kolaydır [20–24].

Farklı rastgele sayı üreteci tasarımlarıyla şifreleme sistemlerinin tasarımı literatürde oldukça yaygındır. Rastgele sayı üreteci tasarımları için gerçek kaynaklı sinyaller ya da tamamen algoritma temelli tasarımlar kullanılabilir. Doğal olarak, bu iki yöntemle üretilen rastgele sayıların çeşitli avantaj ve dezavantajları olacaktır. İstenilen tasarım türüne göre tercih yapılması ve rastgele sayı üreteci türünün seçilmesi gerekir. [25–29].

Rastgele sayı üreteçlerinin değerlendirilmesi yapılırken DIEHARD, TestU01, NIST testleri gibi farklı testler mevcuttur. Genellikle farklı testler sonucunda rastgele sayı üretecinin değerlendirilme süreci farklı olsa da çoğu test benzer parametreleri inceleyerek değerlendirme yapmaktadır.

Bu çalışmada Trivium tabanlı bir sözde rastgele sayı üreteci tasarımı gerçekleştirilmiştir. Gerçekleştirilen tasarım görüntü şifreleme işlemlerinde kullanılmak üzere bir permutasyon üreteci olarak kullanılarak görüntü karıştırma uygulaması yapılmıştır. Yine rastgele sayı üreteci çıktıları kullanılarak gerçekleştirilen şifreleme işlemi üzerinde çeşitli güvenlik testleri yardımıyla değerlendirmeler yapılmıştır.

2. Trivium Algoritması

Trivium algoritmasının rastgele sayı üreteci uygulamalarında yaygın şekilde kullanılmaktadır. Aynı zamanda üretilen rastgele sayılara son-işlem algoritması olarak kullanıldığında rastgele sayı dizisinin istatistiksel özelliklerini geliştirmiştir. Trivium algoritması istatistiksel özellikler açısından kullanılabilir olmasının yanı sıra hızlı işleyen algoritma yapısı ve kolay uygulanabilirlik gibi parametrelerle de öne çıkmaktadır. Düşük gereksinimlere sahip ve mikro ölçeklerdeki sistemlerde bile kullanılabilmektedir. Tablo 1'de Trivium algoritmasının temel parametreleri gösterilmiştir [30–32].

Tablo 1. Trivium için Oluşturulan Parametler

Değişkenler	**Boyut**
Anahtar Uzunluğu	80 bit
Başlangıç Değerinin Uzunluğu	80 bit
İç Durumu Uzunluğu	288 bit

Tablo 1'de gösterilen parametrelerin yanı sıra aşağıda Tablo 2 ve Tablo 3'te verilen sözde kodlar Trivium algoritmasının temel işleyiş düzenini göstermektedir. [30].

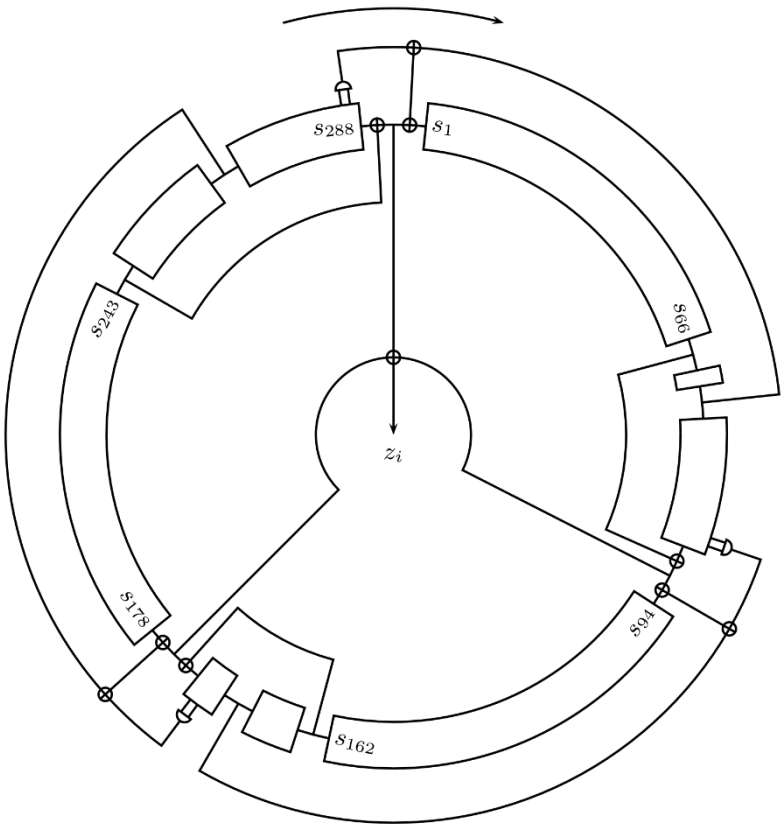
Tablo 2. Trivium Algoritması İç Durum Değişkenlerine İlişkin Sözde Kodlar

$(s_1, s_2, \ldots, s_{93}) \leftarrow (K_1, \ldots, K_{80}, 0, \ldots, 0)$
$(s_{94}, s_{95}, \ldots, s_{177}) \leftarrow (IV_1, \ldots, IV_{80}, 0, \ldots, 0)$
$(s_{178}, s_{179}, \ldots, s_{288}) \leftarrow (0, \ldots, 0, 1, 1, 1)$
for i = 1 to 4 * 288 do
$t_1 \leftarrow s_{66} \oplus s_{91} . s_{92} \oplus s_{93} \oplus s_{171}$
$t_2 \leftarrow s_{162} \oplus s_{175} . s_{176} \oplus s_{177} \oplus s_{264}$
$t_3 \leftarrow s_{243} \oplus s_{286} . s_{287} \oplus s_{288} \oplus s_{69}$
$(s_1, s_2, \ldots, s_{93}) \leftarrow (t_3, s_1, \ldots, s_{92})$
$(s_{94}, s_{95}, \ldots, s_{177}) \leftarrow (t_1, s_{94}, \ldots, s_{176})$
$(s_{178}, s_{179}, \ldots, s_{288}) \leftarrow (t_2, s_{178}, \ldots, s_{287})$
end for

Tablo 3. Trivium Algoritması Bit Üretme Döngüsüne İlişkin Sözde Kodlar

for i=1 to n
$t_1 \leftarrow s_{66} \oplus s_{93}$
$t_2 \leftarrow s_{162} \oplus s_{177}$
$t_3 \leftarrow s_{243} \oplus s_{288}$
$z_i \leftarrow t_1 \oplus t_2 \oplus t_3$
$t_1 \leftarrow t_1 \oplus s_{91} \cdot s_{92} \oplus s_{171}$
$t_2 \leftarrow t_2 \oplus s_{175} \cdot s_{176} \oplus s_{264}$
$t_3 \leftarrow t_3 \oplus s_{286} \cdot s_{287} \oplus s_{69}$
$(s_1, s_2, \ldots\ldots, s_{93}) \leftarrow (t_3, s_1, \ldots.., s_{92})$
$(s_{94}, s_{95,}, \ldots.., s_{177}) \leftarrow (t_1, s_{94}, \ldots.., s_{176})$
$(s_{178}, s_{179,}, \ldots.., s_{288}) \leftarrow (t_2, s_{178}, \ldots.., s_{287})$
end for

Trivium algoritması bir akış şifreleme projesi kapsamında ortaya çıkmıştır. Bit tabanlı çalışan algoritmanın ortaya çıkış amacı en hızlı ve esnek akış şifreleme algoritmasının güvenlik gereksinimlerinden ödün vermeden geliştirilmesi temeline dayanmaktadır. Aynı yarışmada seçilen diğer algoritmalar ise Grain ve MICKEY'dir. Bu algoritmalarla kıyaslandığında Trivium yapısı daha basit tasarımı sayesinde minimum düzeyde devre elemanı kullanmaktadır. Ayrıca çıkış bit oranı daha yüksek olan Trivium daha az güç tüketmektedir [33]. Şekil 1'de Trivium algoritmasının genel çalışma yapısı gösterilmiştir.



Şekil 1. Trivium algoritmasının genel çalışma yapısı [34]

Trivium algoritmasının çalışma mantığı LFSR (Doğrusal Geri Beslemeli Kaydırmalı Kaydedici) mantığına dayanmaktadır. Algoritma içerisinde üç farklı bit uzunluğunda LFSR yapısı mevcuttur. Bu kaydediciler birbirine bağlandığı için dairesel bir yapıdaymış gibi düşünülebilir. Toplamda 288 bit uzunluğuna sahip yapı özel anahtar ve başlangıç vektörüyle işlemlerini başlatır. 80 bitlik başlangıç vektörü ve 80 bitlik gizli işleme sokulduğunda döngüler sonucunda toplamda 2^{64} bit çıkış elde edilebilir.

3. İstatistiksel NIST 800-22 Testi

NIST istatistiksel rastgelelik testleri dünyanın birçok yerinde kabul gören bu alandaki en kapsamlı testlerden biridir. Dünyada bu alandaki standartları belirleyen kurum olan NIST, mevcut şifreleme algoritmalarının standartlarını, güncel kullanım çeşitlerini ve güvenlik açıklarını değerlendiren resmi olarak kabul görmüş bir değerlendirme kuruluşudur. NIST 800-22 istatistiksel rastgelelik testi temelde 15 farklı testten meydana gelmektedir. Tablo 4'de gerçekleştirilen rastgele sayı üreteci tasarımı çıktılarına ilişkin istatistiksel test sonuçları verilmiştir. Her bir test gerçekleştirilirken bir P-değeri hesaplanmaktadır. Bu değer hesaplanırken istatistiksel olarak sıfır hipotez ve eldeki hipotez karşılaştırılarak bir değer elde edilir. İstatistiksel olarak kullanımı farklı olsa da NIST testleri için 0,01 ya 0,001'den büyük değerler başarılı kabul edilmektedir [35].

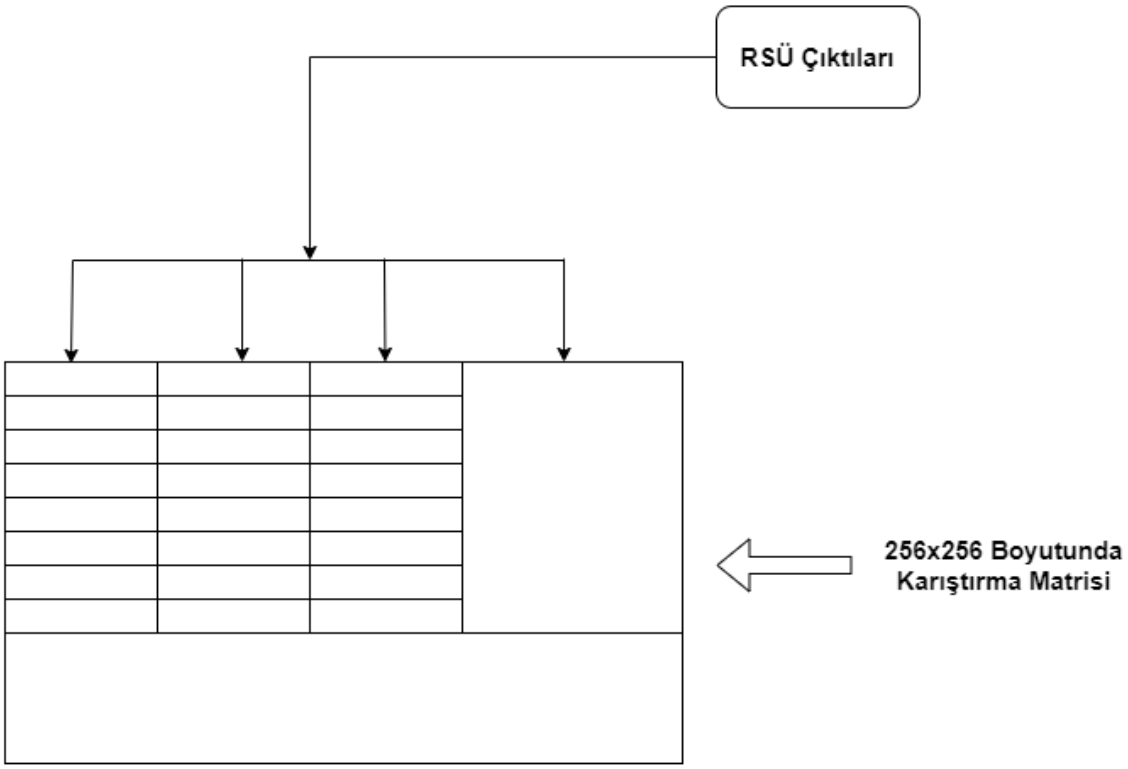
Tablo 4. NIST 800-22 Test Sonuçları

NIST Testleri	Rastgele Sayı Üretecinin Başarımı	Rastgele Sayı Üretecine İlişkin P-Değerleri
Frekans Testi	Başarılı	0,29
Bir Blok içerisinde Frekans Testi	Başarılı	0,51
Akış Testi	Başarılı	0,52
Bir Blok içerisinde en Uzun Akış Testi	Başarılı	0,43
İkili Matris Derece Testi	Başarılı	0,37
Ayrık Zamanlık Fourier Dönüşüm Testi	Başarılı	0,11
Örtüşmeyen Şablon Testi	Başarılı	0,23
Örtüşen Şablon Testi	Başarılı	0,42
Evrensel Test (Maurer Testi)	Başarılı	0,51
Doğrusal Karmaşıklık Testi	Başarılı	0,49
Seri Testi	Başarılı	0,72/0,64
Yaklaşık Entropi Testi	Başarılı	0,33
Birikimli Toplamlar Testi	Başarılı	0,29
Rastgele Gezinimler Testi (Ortalama)	Başarılı	0,42
Rastgele Gezinimler Değişken Testi (Ortalama)	Başarılı	0,37

Tabloya göre NIST 800-22 testlerinden tasarlanan rastgele sayı üreteci başarılı bulunmuştur. Ancak, sadece istatistiksel testlerden başarılı olan rastgele sayı dizilerinin kriptografik uygulamalarda kullanılması uygun olmaz [36]. İstatistiksel başarının yanı sıra rastgele sayı dizileri tahmin edilemez olması ve belirli parametrelere göre geçmiş ya da gelecek sayı dizilerinin başkaları tarafından üretilememesi gerekmektedir [37,38].

4. Piksel Karıştırma ve Görüntü Şifrelemesi Uygulamaları

Şifreleme işlem basamakları bu bölüm altında açıklanacaktır. Öncelikle şifreleme işlemi için 256x256 örnek gri tonlamalı görüntüler kullanılmıştır. Farklı boyuttaki ve nitelikteki görüntüler için geliştirilen algoritmanın uyarlanması mümkündür. Şekil 2'de rastgele permutasyon matrisinin oluşturulma mantığı gösterilmiştir.



Şekil 2. Rastgele permutasyon matrisinin oluşturulması

Burada RSÜ çıktıları alınarak karıştırılmak istenen görüntüyle aynı piksel sayısına sahip bir matris oluşturulur. 256x256 boyutunda yani 1 ile 65536 arasındaki sayılar bu matris içerisine yerleştirilir. Bu işlem için RSÜ çıktıları 16 bitlik sayılara dönüştürülmelidir. Böylece istenilen değer aralığında sayılar üretilmiş olur. Farklı piksel boyutları için buradaki bit sayısının değiştirilmesi yeterli olacaktır. Daha sonra görüntü piksellerine sırasıyla numara verilerek her bir piksel karıştırma matrisi içerisindeki değerle değiştirilir. Örneğin, 1. satır ve 1. sütundaki karıştırma matrisi değeri 256 olsun. Karıştırma işlemi uygulanacak olan görüntüde 1. satır ve 1. sütundaki piksel 256. satır ve 1. sütuna taşınmış olur. Bu şekilde tüm piksellerin yerinin değiştirilmesiyle piksel karıştırma işlemi tamamlanır.

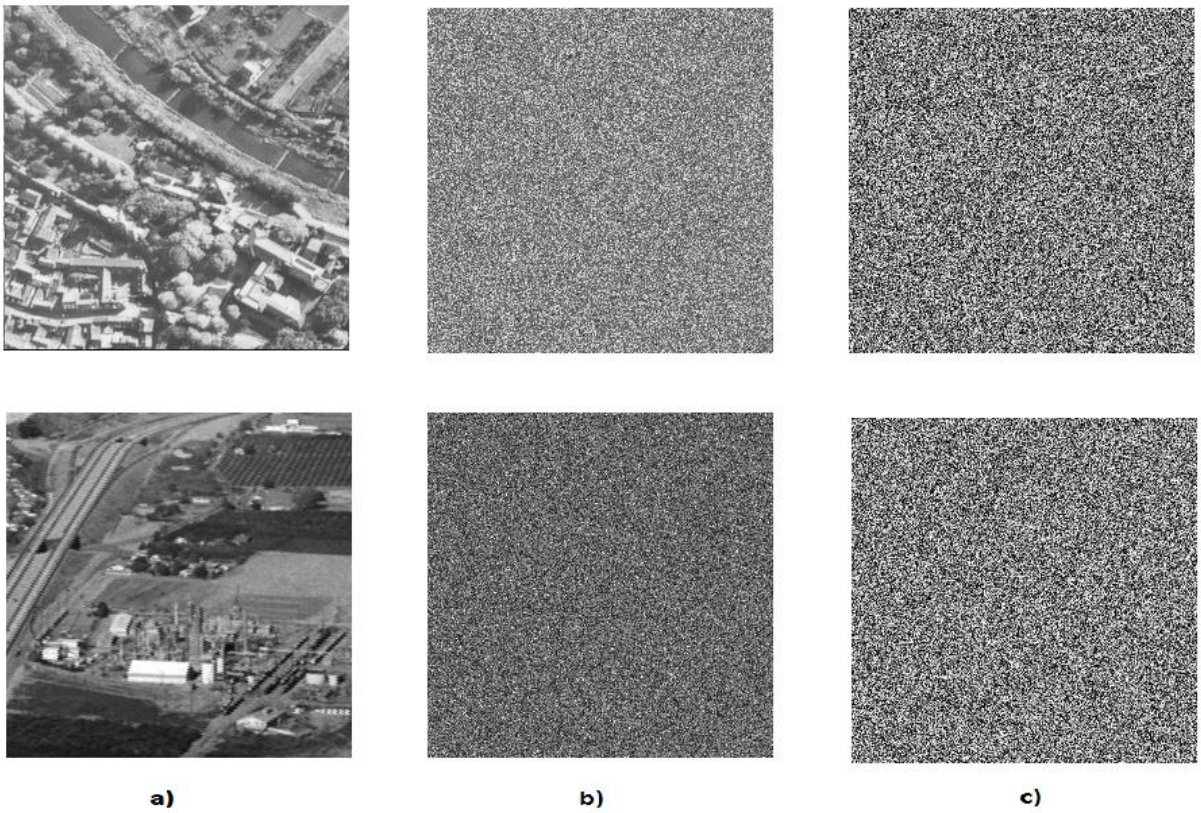
Örnek şifreleme uygulamaları için Vernam Şifreleyicisi kullanılmıştır. En eski şifreleme yöntemlerinden olan Vernam şifreleme yöntemi modern kriptografide güvenilir kabul edilmese de şifreleme için tek kullanımlık orijinal veri boyutunda rastgele sayı dizileri kullanıldığında bu şifreleme sisteminin kırılması mümkün olmamaktadır [39].

Sistemin çalışma sistemi aşağıdaki denklemde gösterilmiştir. Temel olarak şifrelenecek olan verinin her bir bitinin anahtar bitleriyle XOR işlemine tabi tutulmasıyla şifreleme işlemini gerçekleştirmektedir.

$$CipherText = PlainText \oplus Key \tag{1}$$

Burada en önemli nokta şifreleme işlemi için şifrelenecek olan veri ile anahtar olarak kullanılacak RSÜ çıktılarının aynı boyutta olması gerekliliğidir. Ayrıca bu yöntemde kullanılacak olan anahtarın tek kullanımlık olması gerekir. Yüksek boyutlu olmayan veri tiplerinde bu yöntem kullanılabilse de veri boyutu büyüdüğünde şifreleme işlemi için yüksek sistem gereksinimlerine ihtiyaç duyulacaktır. Sistemimizde düşük boyutlu görüntüler kullanıldığı için Vernam şifreleyicisinin kolay uygulanabilir yapısı da göz önüne alınarak bu yöntem tercih edilmiştir.

Yapılan şifreleme uygulaması sonucunda elde edilen çıktılar aşağıdaki şekillerde gösterilmiştir. Piksel karıştırma işlemi ve şifreleme işlemleri sonucunda elde edilen görüntüler ayrı ayrı gösterilmiştir.



Şekil 3. Görüntü şifreleme uygulaması sırasıyla, kuş bakışı şehir manzarası ve kimyasal tesis: **a)** Orijinal görüntü **b)** Piksel karıştırma uygulanan görüntü **c)** Şifrelenmiş görüntü

Görüntü işleme uygulamaları için literatürde sunulan görüntülerden seçilen kuş bakışı şehir manzarası ve kimyasal tesis görüntüleri üzerine başarılı bir şekilde piksel karıştırma ve ardından şifreleme işlemleri uygulanmıştır. Şekilden de anlaşılacağı üzere piksel karıştırma işlemi sonucunda görüntüyü ayırt etmek mümkün olmasa da başta şifrelenen görüntünün piksel yoğunluklarına göre görüntülerde farklılıklar oluşabilmektedir. Ancak şifrelenmiş görüntüler arasında herhangi bir yapısal farklılık gözlemlemek mümkün olmayacaktır.

5. Güvenlik Analizleri

Görüntü şifreleme algoritmalarının güvenilirliğini test etmek için literatürde birçok güvenlik analizi mevcuttur. Güvenlik analizleri yapılmayan bir kriptolojik sistemin saldırılara karşı güvenilir olup olmadığını belirlemek oldukça güçtür.

Anahtar hassasiyeti analizinde şifrelenmiş metin ve şifreleme anahtarı üzerindeki bit oranlarının incelenmesiyle yapılabilir. Aşağıda gösterilen denklem yardımıyla hesaplamaların yapılması mümkündür.

$$T = \frac{n^a}{n^b}\% \tag{2}$$

Burada n^a değeri bit serisi içerisindeki Lojik-1 ve ya Lojik-0 sayılarını göstermektedir. n^b ise toplam bit sayısını göstermektedir. T değeri ise bu değerlerin birbirine yüzdelik olarak oranını vermektedir. Burada bit dizisi içerisindeki Lojik-1 ve Lojik-0 oranları hesaplandığında %50 civarında bir değer bulunması sistemin ideal oranda çalıştığını göstermektedir. Bu oranın %1 oranında değişmesi şifrelenecek metine göre mümkün olabileceği için hesaplanan T değerlerinin %49,5 ile %50,5 arasında olması mantıklı kabul edilir. Mevcut sistem için yapılan hesaplamalarda %49,8 ile %50,2 arasında değerler en yüksek olarak elde edildiği için bu analize göre tasarlanan şifreleme sisteminin başarılı olduğu söylenebilir.

Bilgi entropisi değeri özellikle görüntü şifreleme sistemleri için önemli bir parametre olarak görülmektedir. Aşağıdaki denklem yardımıyla bilgi entropisi değerini hesaplamak mümkündür [40].

$$H_m = \sum_{i=0}^{n} p(m_i) \log \frac{1}{p(m_i)} \tag{3}$$

Formülde hesaplanan H_m entropi değerini $p(m_i)$ ise bir değerin bulunma olasılığını ifade etmektedir. Buradan hesaplanacak olan H_m bilgi entropisi değeri 8 bitlik bir gri tonlamalı resim için ideal şartlarda 8'dir. Bizim geliştirdiğimiz sistemin bilgi entropisi değeri ise 7,99581 olarak hesaplanmış olup ideal değere çok çok yakın olması sebebiyle başarılı olduğu söylenebilir.

Korelasyon katsayıları yine görüntü şifreleme sistemlerinde önemli bir parametredir. Komşu piksellerin korelasyon katsayılarının düşük olması bu görüntünün orijinalinin tespit edilememesini sağlar. Hesaplamalar aşağıdaki formüller yardımıyla yapılabilir [41].

$$r_{x,y} = \frac{E((x - E(x))(y - E(y)))}{\sqrt{D(x)D(y)}} \tag{4}$$

$$E(x) = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{5}$$

$$D(x) = \frac{1}{N} \sum_{i=1}^{N} (x_i - E(x))^2 \tag{6}$$

Burada x ve y değerleri komşu piksellerin yoğunluk değerlerini N ise piksel numaralarını belirtmektedir. Yatay, dikey ve diyagonal korelasyon katsayıları hesaplanırken sadece dikkate alınan piksellerin yönü değişmiş olur. Bu formüller yardımıyla kuş bakışı şehir manzarası görüntüsü için değerler yatay dikey ve diyagonal pikseller yönünde hesaplandığında sırasıyla 0,0064, 0,0198 ve 0,0072 değerleri elde edilmektedir. Elde edilen sonuçlara göre çok düşük bir korelasyonun olduğu görüldüğünden sistemin başarılı olduğu söylenebilir.

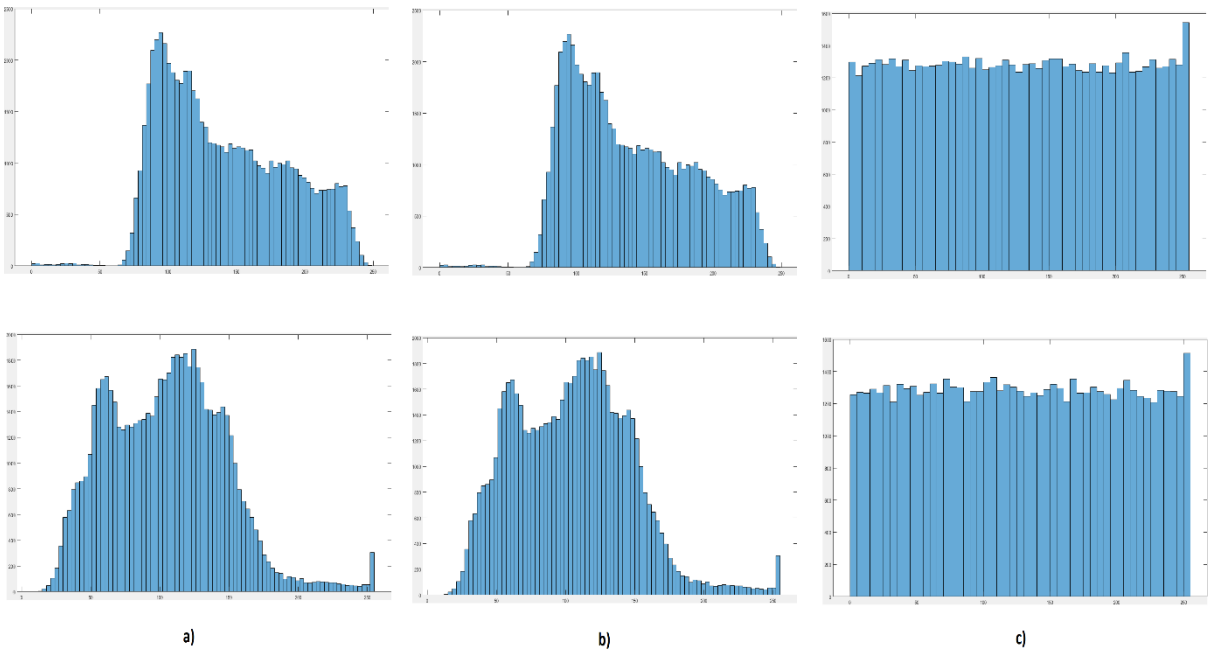
Tablo 5'te yapılan analizlerin literatür ile karşılaştırılması verilmiştir. Önerilen sistemin analizleri yapılırken on farklı şifreleme işlemi sonucunda elde edilen değerlerin ortalaması alınmıştır.

Tablo 5. Test Sonuçlarının Karşılaştırılması

Referans	NIST Testleri	Bit Oranı	Bilgi Entropisi	Korelasyon Katsayıları		
				Yatay	Dikey	Diyagonal
Önerilen	Başarılı	%50,0728	7,99195	0,00174	0,00871	0,00483
[42]	Başarılı	~%50	7,9999	0,022	0,019	0,020
[43]	Başarılı	~%50	7,9973	0,00132	0,00209	-
[44]	Başarılı	~%50	7,99154	0,00186	-0,0015	0,00185
[45]	Başarılı	~%50	7,9973	-0,0029	-0,0032	0,0040

Tabloya göre önerilen sistem ile literatürdeki çalışmaların sobuçları benzerlik göstermektedir. Bu analize göre önerilen sistemin başarılı olduğu söylenebilir.

Aşağıdaki Şekil 4'de şifreleme ve piksel karıştırma işlemleri esnasındaki histogram dağılımları gösterilmiştir.



Şekil 4. Görüntü şifreleme uygulamasındaki histogram dağılımları sırasıyla, kuş bakışı şehir manzarası ve kimyasal tesis: **a)** Orijinal görüntü **b)** Piksel karıştırma uygulanan görüntü **c)** Şifrelenmiş görüntü

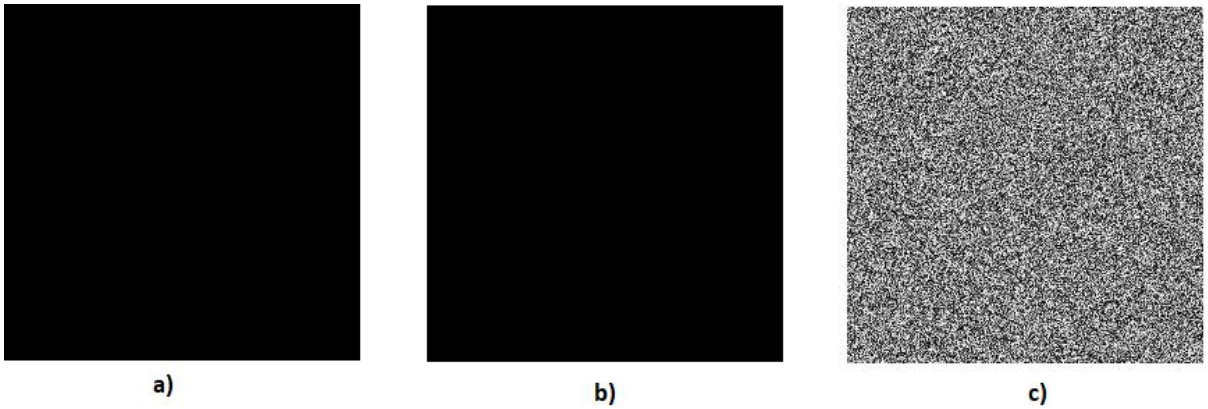
Şekil üzerinde net bir şekilde görüneceği üzere piksel karıştırma işlemi uygulanan görüntülerde orijinal resmi görüntü üzerinden seçmek mümkün olmasa da histogram dağılımı orijinal görüntünün birebir aynısı olduğu için yapılacak analizler yardımıyla görüntünün içeriği hakkında fikir sahibi olunabilir. Bu durum şifreleme işlemi sonrasından ortadan kalktığı için ve şifrelenen her görüntünün benzer bir histogram dağılımına sahip olduğu bilindiği için bu işlemin gerekliliği elzemdir.

Önceki bölümde gösterilen Şekil 3'te piksel karıştırma işlemi uygulanmış ve şifrelenmiş görüntüler gösterilmişti. Piksel karıştırma işlemi sonrasında sadece pikseller yer değiştirdiği için aynı yoğunluklara sahip piksellerin sayısı tamamen aynı kalmaktadır. Bu durum Şekil 4'teki histogram görüntülerinde gösterilmektedir. Tüm şifrelenmiş görüntülerde histogram dağılımları düzgün olduğu için burada orijinal görüntüye ait bir bilgi edinilmesi mümkün olmamaktadır. Piksel karıştırma işleminde ise histogram dağılımları orijinal görüntüyle aynı olduğu için kriptolojik saldırılar neticesinde iletilmek istenen görüntü hakkında fikir sahibi olunabilir.

6. Sonuçlar

Yapılan analizler neticesinde tasarlanan sistemin başarılı bir şekilde görüntü şifreleme uygulamalarında kullanılabileceği ve temel güvenlik gereksinimlerini sağladığı söylenebilir. Piksel karıştırma işleminin sisteme kattığı avantajlar olsa da tek başına yeterli olmadığı sadece piksel karıştırma uygulaması gerçekleştiren çalışmalarda da görülmektedir [46,47]. Tüm şifreleme sisteminin histogram analizinde ve önceki bölümde bahsedilen birçok parametre üzerinde başarılı bir sistem olduğu söylenebilir.

Piksel karıştırma işleminin detaylarının daha iyi anlaşılması için 256x256 piksel boyutundan tamamen siyah bir görüntüye ait veriler piksel karıştırma ve şifreleme işlemine tabi tutularak Şekil 5'te gösterilmiştir.



Şekil 5. Piksel karıştırma ve şifreleme uygulanan siyah görüntü: **a)** Orijinal görüntü **b)** Piksel karıştırma uygulanan görüntü **c)** Şifrelenmiş görüntü

Tamamen siyah piksellerden oluşan bir görüntüye piksel karıştırma işlemi uygulandığında herhangi bir değişim olmadığı şekil üzerinde gösterilmiştir. Ancak şifreleme işlemi uygulandığında görüntünün orijinalinin ne olduğu fark etmeksizin uniform yapıda histogram özellikleri birbirine çok benzeyen görüntüler elde edilmektedir. Böylece piksel karıştırma işlemlerinin tek başına yeterli olmadığı gösterilmiştir.

7. Değerlendirme

Bu çalışmada Trivium algoritması yardımıyla üretilen rastgele sayı dizileri hem permutasyon oluşturularak piksel karıştırma işlemlerinde hem de şifreleme işleminde kullanılmak üzere anahtar üretiminde kullanılmıştır. Üretilen bitlerin rastgelelik koşullarını sağlayıp sağlamadığı NIST testleri yardımıyla başarılı olarak sonuçlandırıldıktan sonra piksel karıştırma ve şifreleme işlemleri uygulanmıştır. Tasarlanan sistemin anahtar hassasiyeti analizi, bilgi entropisi analizi ve korelasyon katsayısı analizlerinde başarılı olduğu belirlenmiştir. Ayrıca histogram analizleri incelendiğinde sadece piksel karıştırmaya dayalı kriptolojik sistemlerin güvenlik açıklarının olduğu, şifreleme işlemlerine olan gereklilik belirtilmiştir.

Yapılan bu çalışmada literatürden farklı olarak Trivium algoritması yardımıyla elde edilen rastgele bir dizileri bir rastgele permutasyon üretmek için kullanılmış ve rastgele permutasyon yardımıyla piksel karıştırma işlemi gerçekleştirilmiştir. Önerilen sistem, karıştırma adımını içermeyen görüntü şifreleme sistemlerine göre bu bağlamda avantaj sağlamaktadır. Karıştırma işlemi sonrası gerçekleştirilen şifreleme işlemiyle önerilen sistemin güvenlik testlerinden başarı sağlaması amaçlanmıştır. Ayrıca karıştırma uygulandığı için şifreleme aşamasının daha esnek şekilde gerçekleştirilebilmesi mümkün olmaktadır.

Yaptığımız çalışmada amaç olarak piksel karıştırma uygulamalarıyla şifreleme işlemlerinin sonucunda oluşan farkları araştırmak hedeflendiği için temel bir şifreleme sisteminin kullanılmasında bir sakınca görülmemiştir. Piksel karıştırma işlemiyle basit düzeyde bir şifreleme sisteminin aralarındaki fark incelenerek şifreleme işlemlerinin gerekliliği gösterilmiştir.

Teşekkür

T. E. fikir sahibi ve deneyleri gerçekleştirdi T. K. sonuçları yorumladı ve düzeltmeleri gerçekleştirdi, T. E. makaleyi yazdı.

Bu çalışma Taha ETEM'in doktora tezinden türetilmiştir.

Kaynaklar

[1] Coşkun S, Pehlivan İ, Akgül A, Gürevin B. A new computer-controlled platform for ADC-based true random number generator and its applications 2019; 27: 847–60.
[2] Etem T, Kaya T. Trivium-Linear Congruential Generator Based Bit Generation For Image Encryption 2020; 32: 287–94.
[3] Moosavi SR, Nigussie E, Virtanen S, Isoaho J. Cryptographic key generation using ECG signal 2017: 1024–31.
[4] Akçay L, Cil E, Vardar A, Yaman I, Yeniceri R, Yalcin ME. Implementation of a chaotic time-delay RNG based secure communication system on FPGA. 2017 10th Int. Conf. Electr. Electron. Eng. ELECO 2017. 2017; 1277–80.

[5] Ayubi P, Setayeshi S, Rahmani AM. Deterministic chaos game: A new fractal based pseudo-random number generator and its cryptographic application 2020; 52: 102472.
[6] Khan JS, Ahmad J. Chaos based efficient selective image encryption 2019; 30: 943–61.
[7] Akgul A, Arslan C, Aricioglu B. Design of an Interface for Random Number Generators Based on Integer and Fractional Order Chaotic Systems 2019; 1: 1–18.
[8] Tuncer T. Implementation of duplicate TRNG on FPGA by using two different randomness source 2015; 21: 35–39.
[9] Kaya T. A true random number generator based on a Chua and RO-PUF: design, implementation and statistical analysis 2020; 102: 415–26.
[10] Li C, Liu Y, Xie T, Chen MZQ. Breaking a novel image encryption scheme based on improved hyperchaotic sequences 2013; 73: 2083–89.
[11] Ceyhan M, Yolaçan EN. Görüntü Dosyalarının Şifrelenerek Güvenli Şekilde Saklanması 2021; 29: 28–42.
[12] Türk S, Şamlı R. Yapay Sinir Ağları İle Klasik Kriptografi Algoritmalarının Şifreli Veriler Üzerinden Sınıflandırılması 2020; 25: 651–64.
[13] Atalay NS, Doğan Ş, Tuncer T, Akbal E. İmge Şifreleme Yöntem ve Algoritmaları 2019; 10: 815–31.
[14] Baykara M, Daş R, Tuna G. A Novel Symmetric Encryption Algorithm and its Implementation 2017: 5–9.
[15] Tuna M, Karthikeyan A, Rajagopal K, Alcin M, Koyuncu İ. Hyperjerk multiscroll oscillators with megastability: Analysis, FPGA implementation and a novel ANN-ring-based True Random Number Generator 2019; 112.
[16] Çimen ME, Kaçar S, Güleryüz E, Gürevin B, Akgül A. Kaotik bir hareket videosunun yapay sinir ağları ile modellenmesi 2018; 20: 23–35.
[17] Çelik H, Doğan N. K-En Az Anlamlı Bitlere Dayalı Kaotik Bir Harita Kullanan Renkli Görüntü Steganografisi 2021: 1–1.
[18] Çavuşoğlu Ü, Akgül A, Kaçar S, Pehlivan İ, Zengin A. A novel chaos-based encryption algorithm over TCP data packet for secure communication 2016; 9: 1285–96.
[19] Akgul A, Kaçar S, Pehlivan I. An Audio Data Encryption with Single and Double Dimension Discrete-Time Chaotic Systems 2015; 5: 14–23.
[20] Zhu L, Song H, Zhang X, Yan M, Zhang T, Wang X, et al. A robust meaningful image encryption scheme based on block compressive sensing and SVD embedding 2020; 175: 107629.
[21] Tiken C, Şamli R. A Comprehensive Review About Image Encryption Methods 2022; 7: 27–49.
[22] Demirtaş M. A Color Image Scrambling Method Based on Zigzag Transform and Cross-channel Permutation 2022; 36: 91–95.
[23] Doğan N, Çelik H. Tarama Modeli Kullanan Karma Bir Görüntü Şifreleme Yöntemi 2021: 1–1.
[24] Balkesn C, Kocer HE. Embedding Encrypted Data into an Image with a Random Pixel Layout Approach 2020: 123–30.
[25] Avaroglu E. Pseudorandom number generator based on Arnold cat map and statistical analysis 2017; 25: 633–43.
[26] Etem T, Kaya T. A novel True Random Bit Generator design for image encryption 2020; 540.
[27] Özkaynak F. Cryptographically secure random number generator with chaotic additional input 2014; 78: 2015–20.
[28] Akgul A, Calgan H, Koyuncu I, Pehlivan I, Istanbullu A. Chaos-based engineering applications with a 3D chaotic system without equilibrium points 2016; 84: 481–95.
[29] Etem T, Kaya T. Self-generated encryption model of acoustics 2020; 170: 107481.
[30] Kaya T. Memristor and Trivium-based true random number generator 2020; 542: 124071.
[31] İnce C, İnce K, Hanbay D. SecureRandom Kütüphanesi Kullanarak Yazılımsal Trivium Oluşturma 2022; 34: 639–44.
[32] Garipcan AM, Erdem E, Tuncer T. Donanım Tabanlı Trivium Akış Şifreleme Algoritmasının FPGA Ortamında Gerçekleştirilmesi 2017; 29: 119–30.
[33] Yun J, Park K-W, Shin Y, Kim H-D. An efficient stream cipher for resistive RAM 2017; 14.
[34] De Canniere C, Preneel B. Trivium Specifications [homepage on the Internet]. n.d. [cited 2022 Jul 1] Available from: (https://www.ecrypt.eu.org/stream/ciphers/trivium/trivium.pdf).
[35] Akkaya S, Pehlivan İ, Akgül A, Varan M. The design and application of bank authenticator device with a novel chaos based random number generator 2018; 33: 1171–82.
[36] Koç ÇK. Cryptographic Engineering. Springer US 2009.
[37] Yakut S, Tuncer T, Ozer AB. Secure and efficient hybrid random number generator based on sponge constructionsfor cryptographic applications 2019; 25: 40–46.
[38] Yakut S, Tuncer T, Özer AB. A New Secure and Efficient Approach for TRNG and Its Post-Processing Algorithms 2020; 29.
[39] Özkaynak F. Kriptolojik Rasgele Sayı Üreteçleri 2015; 8: 37–44.
[40] Chen X, Qian S, Yu F, Zhang Z, Shen H, Huang Y, et al. Pseudorandom Number Generator Based on Three Kinds of Four-Wing Memristive Hyperchaotic System and Its Application in Image Encryption 2020; 2020.
[41] Sang Y, Sang J, Alam MS. Image encryption based on logistic chaotic systems and deep autoencoder 2022; 153: 59–66.
[42] Abutaha M, Amar I, Alqahtani S. Parallel and Practical Approach of Efficient Image Chaotic Encryption Based on Message Passing Interface (MPI) 2022; 24: 566.
[43] Gürevin B, Yıldız M, Güleryüz E, Ç Kutlu M, Sorgun Ö. A Chaos Based Image Encryption On LabVIEW 2020; 2.
[44] Kadir A, Aili M, Sattar M. Color image encryption scheme using coupled hyper chaotic system with multiple impulse injections 2017; 129: 231–38.

[45] Chai X, Fu X, Gan Z, Lu Y, Chen Y. A color image cryptosystem based on dynamic DNA encryption and chaos 2019; 155: 44–62.
[46] Dursun G, Özer F, Özkaya U. A new and secure digital image scrambling algorithm based on 2D cellular automata 2017; 25: 3515–27.
[47] Diaconu AV, Costea A, Costea MA. Color image scrambling technique based on transposition of pixels between RGB channels using Knight's moving rules and digital chaotic map 2014; 2014.
[48] Tanyıldızı E, Orhan A. An introduction to variable and feature selection. Comput Appl Eeng Educ 2009; 17(2): 187-195.
[49] Duman M, Gürbüz AC. 3D imaging for ground-penetrating radars via dictionarydimension reduction. Turk J Elec Eng & Comp Sci 2015; 23(5): 1242-1256.
[50] Haupt RL, Haupt SE. Practical Genetic Algorithms. 2nd ed. New York, NY, USA: Wiley, 2004.
[51] Kennedy J, Eberhart R. Swarm Intelligence. San Diego, CA, USA: Academic Press, 2001.
[52] Poore JH, Lin L, Eschbach R, Bauer T. Automated statistical testing for embedded systems. In: Zander J, Schieferdecker I, Mosterman PJ, editors. Model-Based Testing for Embedded Systems. Boca Raton, FL, USA: CRC Press, 2012. pp. 111-146.
[53] Li RTH, Chung SH. Digital boundary controller for single-phase grid-connected CSI. In: IEEE 2008 Power Electronics Specialists Conference; 15-19 June 2008; Rhodes, Greece. New York, NY, USA: IEEE. pp. 4562-4568.
[54] Boynukalın Z. Emotion analysis of Turkish texts by using machine learning methods. MSc, Middle East Technical University, Ankara, Turkey, 2012.