

# Stokastik Aritmetik-Lojik Birim Tasarımı ile Görüntü İşlemede Sobel Filtre Operasyonu

## Sobel Filter Operation in Image Processing via Stochastic Arithmetic-Logic Unit Design

Sercan Aygün<sup>1,2</sup>, Mustafa Altun<sup>2</sup>, Ece Olcay Güneş<sup>2</sup>

<sup>1</sup>Bilgisayar Mühendisliği Bölümü, Yıldız Teknik Üniversitesi, Davutpaşa, İstanbul, Türkiye

<sup>2</sup> Elektronik ve Haberleşme Mühendisliği Bölümü, İstanbul Teknik Üniversitesi, Maslak, İstanbul, Türkiye  
sercan@ce.yildiz.edu.tr, altunmus@itu.edu.tr, ece.gunes@itu.edu.tr

### I. GİRİŞ

**Özetçe**—Teknolojik gelişmelerin artması ile birlikte ortaya çıkan işlemcilerin hız sınırı, nano teknolojik çözümlerin transistörleri nanometreler mertebesinde birkaç atom boyutuna indirmesini ivmelendirmiştir. Atom davranışından hareketle deterministik hesaplamaların yerine stokastik hesaplamaların kritik olduğu fiziksel sistemler ele alınmaktadır. Bu sebeple, disiplinler arası bir çalışma olarak bu bildiriye stokastik davranıştaki aritmetik-lojik birimin tasarımı ve görüntü işlemeye uygulanmasından bahsedilecektir. Tasarım kısıtları göz önüne alınarak yapılan lojik seviyesi tasarım ardından, bilinen görüntü süzgeç operasyonları sırasında kullanılacaktır. Aritmetik lojik birimin girişine gönderilen ikili hale getirilmiş olasılıksal piksel değerleri Sobel görüntü işleme işlecine tabi olacaktır. Görüntüler hem geleneksel süzgeç hem de stokastik süzgeç ile işlenip elde edilen sonuçlar görsel olarak verilecektir.

**Anahtar Kelimeler**—ALU tasarımı; görüntü işleme; hesaplamalı nanoelektronik; nanoteknoloji; Sobel işleci; stokastik hesaplama.

**Abstract**—With the increase of technological developments, the speed limit of the processors that have emerged has accelerated the reduction of nanotechnological solutions of transistors into atomic dimensions from nanometers. Physical systems in which stochastic computation is critical instead of deterministic computation by taking action from atomic behavior are discussed. For this reason, as an interdisciplinary study, this paper will discuss the application of the arithmetic-logic unit in stochastic behavior for the design issues and image processing applications. The logic level design made by considering the design constraints will be used during known image filtering operations. Probabilistic pixel values in binary that sent to the input of the arithmetic logic unit will be used in the Sobel image processing operator. The images are processed with both conventional filter and stochastic filter, then the results obtained will be given visually.

**Keywords**—ALU design; image processing; computational nanoelectronics; nanotechnology; Sobel operator; stochastic computing.

Nanoteknoloji tabanlı elektronik devre tasarımı ve uygulamaları geleceğin teknolojileri arasında yer almaktadır. İşlemcilerin hız sınırına yaklaşması ve Moore yasasının öngörüsü ile fiziksel anlamda transistör boyutundaki küçülme, geleceğin teknolojilerini/tasarımlarını atom boyutu seviyesine indirmeyi gerektirmektedir. Nanoteknolojik tasarımlar; diyot tabanlı, 2 kapılı CMOS tabanlı, 4 kapılı nano dizinler gibi literatüre kazandırılmış olan temel elektronik elemanlarıyla tasarlanan yapılarıdır [1][2]. Bu yapılar ile tamamlayıcı lojik (*complementary logic*) için geçerli olan tasarım yöntemleriyle lojik fonksiyonlar gerçekleştirilebileceği için oldukça kullanışlı ve nanoteknolojik çözümler elde edilmektedir. Nano mertebesindeki elektronik devrelerin yalnızca boyut açısından atom taneleri mertebesinde olması değil, davranışsal olarak da atom tanecikleri gibi olasılıksal süreçler kapsamında incelenmesi gereklidir. Bu, kullanılan malzemelerin doğası ile ilgilidir ve kuantum bilgisayarları için önemli bir özellik olarak verilerin ikili  $\{0,1\}$  durumunun bir olasılıkla ifade edilmesidir. Bu çalışmada disiplinler arası bir yaklaşım ile lojik seviyesi donanımsal tasarımın olasılıksal kapılar ile tasarlanıp ardından bir uygulama olarak görüntü işlemede Sobel işleci için kullanılması öngörülmektedir. Tasarlanan ALU (Aritmetik Lojik Birim) ile işlemsel operasyonlar gerçekleştirilecek, ancak kullanılan lojik kapılar deterministik değil olasılıksal davranışta olacaktır.

Literatüre bakıldığında Sobel işlecinin donanımsal olarak gerçekleştirilmesi çoğunlukla FPGA üzerinde yapılmıştır. Koyuncu ve ark. bir kenar belirleme uygulaması olarak FPGA üzerinde Sobel işlecini çalıştırmışlardır [3]. Chaple ve ark. Robert, Prewitt, Sobel işlecini gerçek zamanlı olarak FPGA üzerinde [4], yine Khairnar ve ark. Sobel uygulamasını Verilog ile gerçekleştirmişlerdir [5]. Öte yandan, Alghurair ve Al-Rawi tasarladıkları ALU'yu kullanarak FPGA üzerinde Sobel süzgecini kullanmışlardır [6]. Çoğu çalışmanın sahada programlanabilir kapı dizileri (FPGA) üzerinde tamamlanması, bu yönü ile çalışmamızı nano dizinlere hazır tasarlanabilir bir model oluşturması açısından stokastik yapısı ile de farklı kılmaktadır.

## II. KISITLAR VE TASARIM

Bu bölümde tasarıma ilişkin belirlenmiş kısıtlar ve lojik seviye tasarıma ait detaylar verilecektir. Tasarlanacak ALU devresinin A ve B iki girişi olmak üzere bu girişler  $A > B$  olacak şekilde 4 bit kesirli sayılar  $(0.x_1x_2x_3x_4)_2$  formatında devre girişine uygulanmaktadır. Yani,  $(0.0000)_2 = 0/16$ ,  $(0.0001)_2 = 1/16$ ,  $(0.0010)_2 = 2/16, \dots, (0.1110)_2 = 14/16$ ,  $(0.1111)_2 = 15/16$  ikili ifade edilmiş kesirler dikkate alınmaktadır. ALU'nun S seçim biti ortalama toplama ve çıkarma işlemleri için kullanılacak; S=0 için  $(A-B+1)/2$  ve S=1 için  $(A+B)/2$  işlemleri yapılacaktır. Örneğin, S=0 için  $A=0.1010$  ve  $B=0.1001$  olmak üzere sonuç  $Y = 0.00001 = (0.1010 - 0.1001)/2$  elde edilecektir. Öte yandan olasılıksal olarak elde edilen sonuçlara bağlı ortaya çıkan hata için;

$$Hata = \frac{yuvarla(|32 \times (z - z_{beklenen})|)}{32 \times z_{beklenen}} \quad (1)$$

denklemini kullanılacaktır. Yuvarlama işlemi en yakın tamsayıya olacak şekilde yapılacaktır. Hata hesabı için örneğin,  $A = 0.1011 = 11/16$  ve  $B = 0.0011 = 3/16$  olmak üzere,  $S = 1$  ( $(A+B)/2$  için) ortalama işlemi için, z çıkış olasılık değeri,  $14.4/32 = 0.45$  olarak elde edilecektir. Bu değer 1'lerin dizi içindeki durumuna göre gerçek sonucu vermektedir. Öte yandan  $z_{beklenen}$  çıkış olasılık değeri idealdeki olasılık değeri olarak bulunan; bit dizisinin akış sırasındaki 1'lerin dizilme durumu düşünülmeden, yalnızca olasılık değerlerinin nümerik hesabı ile bulunan değerdir. Verilen örnekte beklenen değer,  $z_{beklenen} = 14/32$  olmak üzere hata  $0.4/14 = 0$  olarak bulunur.

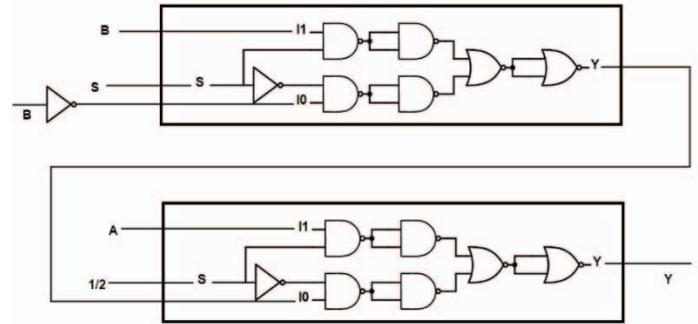
Donanımsal tasarım sırasında uyulması öngörülen kısıtlar veya diğer deyişle isteri analizi, maliyet de göz önüne alınarak belirlenmiştir. Hesaplama zamanı açısından maliyet fonksiyonu da şu şekilde verilmektedir:

$$Hesaplama\ zamanı\ maliyet\ fonksiyonu = (giriş\ sekansı\ bit\ sayısı) \times (lojik\ kapıların\ sayısı) \quad (2)$$

Yukarıdaki formülde görülmektedir ki giriş bit uzunluğu ve kullanılan lojik kapı sayısı hesaplama zamanını arttırmaktadır. Bu sebeple kullanılacak olan algoritmaya bağlı bit dizilerinin uzunluğu ve tasarlanan donanımın en az sayıda eleman içermesi, başarımlı açısından oldukça önemlidir. Tasarım sırasında mantık devresinin fiziksel anlamda sahip olacağı en uygun sayıdaki kapı sayısı, bazı azaltma yaklaşımlarından sonra elde edilebilir. Boole cebri veya Karnaugh diyagramı ile en az sayıda kapı kullanımı sağlanabilir. Bu kapı sayısı azaltma işleminden sonra, giriş bitleri de kullanılarak maliyet kontrol edilmeli ve doğruluğa bakılarak hata analizi yapılmalıdır. Hesaplama zamanını da etkileyen maliyet fonksiyonu en iyi değere getirilmelidir. edilmelidir. Bu çalışmada tasarımlar sırasında yalnızca iki girişli NAND (*Ve-Değil*), NOR (*Veya-Değil*) ve NOT (*Değil*) kapıları kullanılacaktır. NAND ve NOR kapıları evrensel kapılardır; yani yalnızca bu kapılar ile diğer tüm lojik operasyonlar gerçekleştirilebilmektedir.

## A. Aritmetik-Lojik Birimin Tasarımı

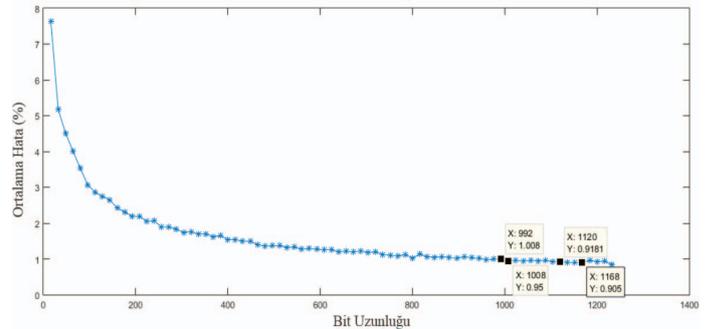
Bu bölümde, kullanılacak olan ALU'nun tasarımından bahsedilecektir. Toplama ve çıkarma temelli çalışacak bu birim bu çalışma kapsamında süzgeç işlemleri sırasında toplayıcı (*adder*) ve çıkarıcı (*subtractor*) görevi görecek olup, ileriki çalışmalarda karşılaştırıcı (*comparator*), kaydırıcı (*shifter*) gibi başka operasyonları da kullanarak özellik çıkarımı algoritmalarına uygulanabilecektir. Önceki bölümde bahsedildiği üzere ALU tasarımı için yalnızca 3 farklı tipte lojik kapı kullanılacaktır. Şekil 1'de bu çalışma kapsamında tasarlanan devreye ilişkin detaylar sunulmaktadır. Burada birbirine kaskat bağlı iki adet 2-girişli-1-çıkışlı MUX (*multiplexer-çoklayıcı*) yapısı bulunmaktadır. Tek bitlik seçim biti S'nin durumuna göre, {0,1},  $(A+B)/2$  veya  $(A-B+1)/2$  işlemlerinden biri gerçekleştirilecektir. Altta ikinci MUX'un seçme biti olarak 1/2 değeri bir olasılık değeri olarak girişe gönderilmektedir. Tasarımda yalnızca *Ve-Değil*, *Veya-Değil* ve *Değil* kapıları kullanılmıştır ve maliyet için en uygun sayıda kapı bulunmaktadır.



Şekil 1. Tasarlanan sayısal devre: MUX tabanlı ALU.

## B. En Uygun Hata

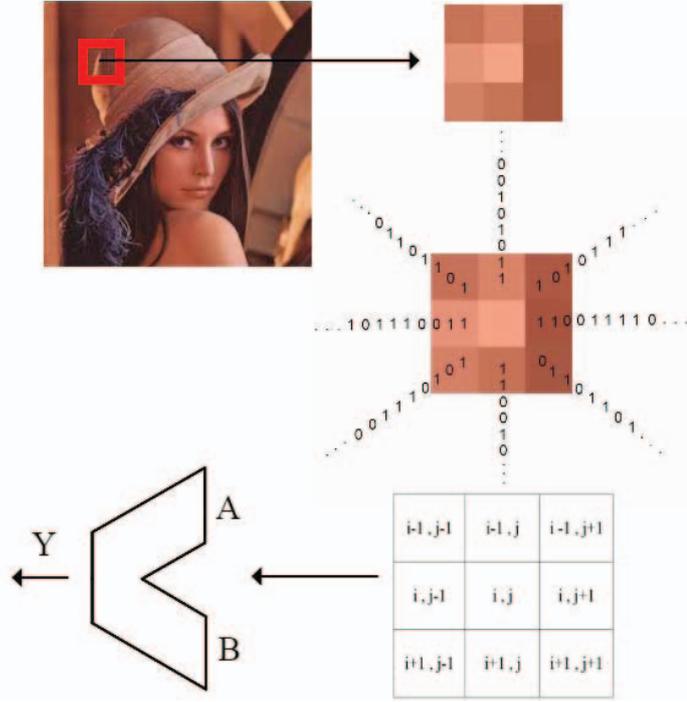
Öte yandan ALU devresinin %1 gibi bir hata oranı için gerekli bit uzunluğu kontrol edilmiştir. Sonsuz uzunluklu girişler %100 doğruluk oranı ile hesaplama sağlar ancak zaman parametresi ve uygulanabilirlik açısından pratikte bu durum söz konusu değildir. Uzun paketlerin girişe gönderilmesi hatayı azaltacaktır. 1 numaralı denklemde verilen hata formülü uyarınca doğrusal olarak artan bit uzunluğuna ilişkin devrenin verdiği ortalama hatanın grafiği Şekil 2'de verilmektedir. Sonuç olarak yaklaşık toplam 992-1008 civarında giriş biti sayısı ile ~%1 hata ortaya çıkmaktadır.



Şekil 2. Hata azaltma için en uygun bit uzunluğu grafiği.

### III. MOTİVASYON VE YÖNTEM

Bu bölümde görüntü işleme de kullanılan yaklaşımdan ve lojik kapıların olasılıksal davranışından bahsedilecektir. Şekil 3’de görsel olarak işleyişe ilişkin bir akış verilmiştir. Bir süzgeç gibi çalışan 3x3 boyundaki işleç için pikseller sırası ile görüntüde taranarak evrişim (convolution) operasyonu yapılmaktadır. Sobel operasyonu olan bu işlem ile ALU üzerinde tanımlanmış aritmetik işlemlerden faydalanarak, girişe piksel değerlerine bağlı olarak gelen bit dizileri ikili hesaplanacak ve çıkış dizisi yeni piksel değeri olarak belirlenecektir.



Şekil 3. Görüntü işleme için her piksele atanan olasılıksal bit dizileri ve süzgeç işlemleri için ALU kullanımı.

#### A. Sobel İşleci

Pikseller arası gri seviyesi değerlerinin değişimine göre x-yönünde ve y-yönünde analiz yapan Sobel işleci, 3x3 bir süzgeç olarak iki yönlü türev işlemi yapan Şekil 4’de sunulmuş bir operatördür. Görüntüye uygulanan bu evrişim ile yataydaki ve dikeydeki eğimler, *gradient*, tespit edilir. Yatay ve dikey türevler I görüntüsü Şekil 4’deki yapılar ile evrişim için hesaplandıktan sonra  $G_x$  (*Gradient-x*) ve  $G_y$  (*Gradient-y*) hesaplanır. Buradan;

$$G = \sqrt{G_x^2 + G_y^2} \quad (3)$$

ifadesi ile eğim ( $G$ ) elde edilir. 3 numaralı denklem, iki boyutlu görüntü piksellerinin yatay ve dikey yönlü türevlerini içeriyor olup,  $f(x,y)$  yoğunluk fonksiyonu olmak üzere iki boyutta eğim;

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (4)$$

olarak bulunur ve buradan da büyüklük (*magnitude*) değeri 5 numaralı denklem ile hesaplanarak;

$$|\nabla f| = \sqrt{\left[\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right]} \quad (5)$$

3 numaralı denklemdeki ifade elde edilir [3] [7] .

$$I^* = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (a) \quad I^* = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (b)$$

Şekil 4. Sobel işleci için x ve y düzleminde uygulanan operatörler ( $G_x$  ve  $G_y$  hesaplanması).

#### B. Olasılıksal Lojik Kapılar

Geleneksel lojik kapıların davranışı bit tabanlı olarak girişlere gelen verinin, lojik operasyona tabi tutulması şeklindedir. Giriş(ler), lojik kapı(lar) ve çıkış(lar) deterministik olmak üzere, beklenen ve tahmin edilen sonuçlar elde edilir. Nanoelektronikte stokastik davranış gösteren lojik kapılar bit dizisi üzerinden işlem yapar. Tasarım için kullanıldığında belirli bir olasılık ile dizi şeklinde gelen giriş verilerinin çıkışa tek tek ikili olarak işlenmesi şeklinde çalışan kapılara bir örnek Şekil 5’de verilmektedir. Bağımsız olarak her bitin  $\{0,1\}$  olma olasılıkları toplam giriş P olasılığını değiştirmese de, ortaya çıkış sırasına bağlı olarak beklenen  $P_{\text{çıkış}}$  değerini bir hata ile etkileyebilir. İkili sekans uzunluğunun her biti için 1 gelme olasılığı Binom dağılımı olarak alınırsa, şekilde  $\frac{4}{8}$  olasılık ve  $\frac{6}{8}$  olasılıkla ortaya çıkmış olan ikili sekanslar çıkışa  $\frac{4}{8} \times \frac{6}{8} = \frac{3}{8}$  olasılıkla; yani 8 bitlik paket içinde 3 tane (1)<sub>2</sub> değeri olacak şekilde çıkış verir. Ancak görüldüğü gibi beklenen ve ortaya çıkan olasılık değerleri farklı olabilmekte; bu da önceki bölümde anlatılan hata kavramını ortaya koymaktadır [8][9][10].



Şekil 5. Ve kapısı için ikili dizilerin girişe gönderimi ve olasılıksal davranışı.

Bu kapsamda görüntü işlemenin rasgele bir süreç olarak incelenmesi, her pikselin bağımsız bir Binom dağılımı olduğu fikrine dayanmaktadır. Stokastik süreçte her pikselin kendi dağılımı vardır. Bunun için her piksel n bitlik uzunluk için bağımsız olarak n farklı olasılık kullanılır. Her bağımsız çaba aşlında Bernoulli dağılımı olup iki olası sonuç, 0 veya 1 ile giriş dizisi yaratılmaktadır. Binom dağılımı, bağımsız ve aynı şekilde dağıtılan Bernoulli rasgele değişkenlerinin toplamıdır. Binom olasılığı için 6 ve 7 numaralı denklemler verilebilir.

$$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k} \quad (6)$$

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \quad (7)$$

Her piksel,  $n$  bağımsız olasılıklı Binom dağılımına sahiptir. Bu nedenle, her görüntü pikseli  $G$  gri seviyeye sahip olup  $p = G / 2^{m\text{-bit gri seviyesi}}$  olasılıkla 1'lerle dolu bit sekansına sahip olacak ve test amacı ile  $n = 510$  seçilerek *Binom* ( $n, p$ ) hesaplanacaktır. Sonuçta, 510 bağımsız denemede, bağımsız olarak kaç tane 1'in görüldüğü bir rastlantısal olay olarak ortaya çıkacak ve beraberinde bir hata getirecektir. Bu çalışmada Sobel işleci için stokastik ALU'dan toplama ve çıkarma işlemleri için kullanılacak ve girişine her bir görüntü pikseline ilişkin Binom olasılığına bağlı olarak elde edilen bit akışı verilecektir. ALU'nun iki girişine gelen her bit, ikili olarak lojik operasyonların ardından çıkışa yazılarak çıkışta da sonuç sekansı  $n$ -bit uzunluğunda elde edilecektir. Piksellere bağlı olasılık değerlerine göre belirlenen bit akışları, 1 ve 0'ların  $n$ -bit uzunluğunda nasıl dağılım göstereceği tamamen Binom dağılımına bağlıdır ve olasılıksaldır.

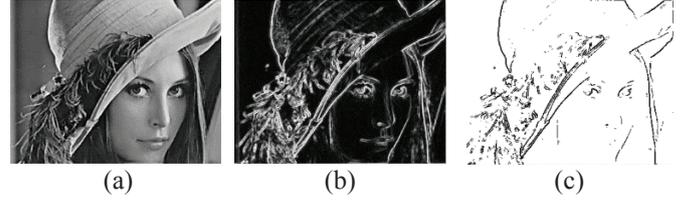
#### IV. ANALİZ, TEST VE SONUÇLAR

Bu bölümde elde edilen sonuçlara ilişkin bilgiler verilecektir. Bu çalışmada nanoelektronik çözümlerin uygulanabilirliği açısından önemli bir tasarım modülü olan ALU tasarımı ve uygulandığından bahsedilmiştir. İkili girişinde 4-bit kesirli sayıları olan ALU ile görüntü piksellerine bağlı elde edilen bit sekansları devrenin girişine verilmiş ve çıkış sekansı tekrar piksel değeri olarak ifade edilerek yeni görüntü elde edilmiştir. MUX yapısı kullanılarak tasarımı yapılan devre için %1'lik bir hatada 998 ve üzeri bit uzunluğu gerekli olduğu görülmektedir. Binom dağılımındaki  $n$  sayısı, yani bit uzunluğu, ne kadar büyük olursa o ölçüde daha doğru sonuçlar elde edilir. Teoride sonsuz uzunluktaki bit diziliminin %100 doğruluk vermesi beklenir. Ancak  $n$  arttıkça, hesaplama zamanı da beraberinde artmaktadır. Zaman ve başarımlı çelişkisi uygulamaya özgü olarak değerlendirilerek hangisinin kritik olduğuna bağlı olarak karar verilmelidir.

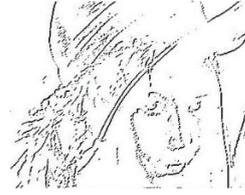
Şekil 6 ve 7'de, geleneksel (olasılıksal yapıda olmayan sıradan Sobel operasyonu) ve stokastik çalışan Sobel sonuçları sunulmuştur. Geleneksel ve stokastik görselin özellikleri oldukça yakındır. Teknik olarak görüntülerin arasındaki fark/benzerlik PSNR (*Peak Signal to Noise Ratio*) yani sinyal gürültü oranı ile kontrol edilmektedir. Sinyal gürültü oranı dB mertebesinde ve 50-70 dB'lik bir sinyal gürültü oranı iyi bir görüntü benzerliği olarak kabul edilebilir. Olasılıksal görüntü işleme ve donanım üzerinde uygulaması söz konusu ise 20 dB veya daha yüksek bir sinyal gürültü oranı, görüntü yakınlığının iyi olduğunu gösterir. İki özdeş görüntü sonsuz sinyal/gürültü değeri verir. Bu nedenle, sinyal gürültü oranı arttıkça iki görüntü arasındaki benzerlik de artar. Bildiri kapsamında tasarlanan olasılıksal ALU üzerinde uygulanan Sobel işleciyle süzülen görüntü ve geleneksel Sobel ile kenar bulunmuş görüntü kıyaslandığında sinyal gürültü oranı yaklaşık olarak 20 dB mertebesinde elde edilmiştir.

Stokastik süzgeç donanım maliyeti açısından klasik yaklaşıma göre çok daha iyi sonuçlar vermektedir. Çünkü daha küçük alan avantajını söz konusudur. Güç tüketimi de düşük alandan ötürü daha az olmaktadır. Ancak stokastik süzgecin hesaplama süresi geleneksel yaklaşıma göre daha kötüdür. Buradan alan-gecikme ikilemi doğmakta; birinden kazanç

diğerinden kayıp getirmektedir. Stokastik süzgecin özellikle düşük alan düşük güç gerektiren mobil, IOT benzeri sistemlerde öne çıkması gelecek çalışmalar açısından da oldukça önü açık bir araştırma alanı olduğunu göstermektedir.



Şekil. 6. Örnek bir (a) görüntüsü için, gradient ile elde edilen Sobel (geleneksel) (b) ve eşikleme uygulanmış kenarlar (c) .



Şekil. 7. Stokastik Sobel işleci uygulanmış görüntü.

#### KAYNAKLAR

- [1] D. B. Strukov ve K. K. Likharev, "Reconfigurable nano-crossbar architectures (kitap bölümü)," *Nanoelectronics and Information Technology*, R. Waser, Ed., 3rd ed. Hoboken, NJ, USA Wiley, 2012.
- [2] M. Altun ve M. D. Riedel, "Logic Synthesis for Switching Lattices," *IEEE Transactions on Computers* 61(11) pp. 1588-1600, 2012.
- [3] İ. Koyuncu, Ö. Çetin, F. Katircioğlu ve M. Tuna, "Edge detection application with FPGA based Sobel operator," *2015 23rd Signal Processing and Communications Applications Conference (SIU)*, Malatya, 2015, pp. 1829-1832.
- [4] G. N. Chaple, R. D. Daruwala ve M. S. Gofane, "Comparisons of Robert, Prewitt, Sobel operator based edge detection methods for real time uses on FPGA," *2015 International Conference on Technologies for Sustainable Development (ICTSD)*, Mumbai, 2015, pp. 1-4.
- [5] T. Khairmar, Harikiran, A. Chandgude, S. Sivanantham ve K. Sivasankaran, "Image edge detection in FPGA," *2015 Online International Conference on Green Engineering and Technologies (IC-GET)*, Coimbatore, 2015, pp. 1-4.
- [6] D. Alghurair ve S. S. Al-Rawi, "Design of Sobel operator using Field Programmable Gate Arrays," *2013 The International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAECE)*, Konya, 2013, pp. 589-594.
- [7] E. Aybar, "Sobel İşleci Kullanılarak Renkli Görüntülerde Kenar Bulma," *Afyon Kocatepe Üniversitesi Fen Ve Mühendislik Bilim. Derg.*, vol. 8, no. 1, pp. 205-217.
- [8] R. R. Amalraj, R. Joshi ve G. Vishwanathan "A comparison of deterministic and stochastic logic for image processing applications," *University of Wisconsin - Madison Dept. of Electrical & Computer Engineering, ECE 753 Project*.
- [9] H. Chen ve J. Han, "Stochastic computational models for accurate reliability evaluation of logic circuits". *20th symposium on Great lakes symposium on VLSI (GLSVLSI '10)*. ACM, New York, NY, USA, pp. 61-66.
- [10] S. Yavuz ve M. Altun, "Stokastik hesaplamada hata oranlarını azaltmak için rastgele bit karıştırma yöntemi," *ELECO*, Bursa 2014.