

FPGA Donanımı Üzerinde Çalışan Elma Sınıflandırma ve Boyutlandırma Sistemi

Eser SERT

Kahramanmaraş Sütçü İmam Üniversitesi, Bilgisayar Mühendisliği Bölümü, Kahramanmaraş, Türkiye
esersert80@gmail.com

(Geliş/Received: 24.02.2018; Kabul/Accepted: 07.08.2018)

Özet

FPGA (Field Programmable Gate Array) teknolojisi ile görüntü işleme, veri madenciliği, nesne tanıma gibi uygulamalar yüksek performansta gerçekleştirilebilmektedir. FPGA'daki üstün mimari yapısı sayesinde görüntü işleme uygulamalarında çok hızlı şekilde sonuçlar alınabilmektedir. Bu donanım bilgisayardan bağımsız olarak çalıştığından yapılan çalışmanın mobilleşmesi sağlanmaktadır, bu ve benzeri avantajlarından dolayı çalışmada FPGA tercih edilmiştir. Tasarlanan sistemle gerçek zamanlı olarak granny smith (GS) ve starking (ST) tür elmaların genişlik ölçümü ve renge bağlı olarak sınıflandırma işlemleri yapılmaktadır. Çalışmada, segmentasyon ve sınıflandırma işlemlerinde kullanılan en başarılı danışmansız öğrenme yöntemlerinden birisi olan K-means kümeleme metodu tercih edilmiştir. Bu metod çalışmamıza uygun şekilde modifiye edilerek VHDL (Very High Design Language) diliyle FPGA'da uygulanmıştır. Tasarlanan sistem üzerinde yapılan test işlemlerinde 100 adet elma için ortalama % 2.09 bağıl hata ile elma genişliği ölçülmüş ve %94 başarı ile elma sınıflandırma işlemi gerçekleştirilmiştir.

Anahtar Kelimeler: FPGA, VHDL, K-means, Elma Sınıflandırma

Apple Classification and Dimensioning System Based that Runs on FPGA Hardware

Abstract

Thanks to FPGA technology, applications such as image and video processing, data mining and pattern recognition can display better performance. Its superior hardware features can help reach quick and accurate results in these applications. In addition, the designed system can operate in mobile mode as because FPGA hardware is a standalone processing unit, making FPGA favorable for this study. In this study, K-means clustering algorithm was used because it is one of the most successful unsupervised learning algorithms in the segmentation and classification of granny smith (GS) and starking (ST) apples. After the algorithm was modified in a suitable way to our study, the algorithm was applied to FPGA hardware in VHDL (Very High Design Language) language. The designed system measures the width of apples and classifies them based on their color in real time. The designed system tested with 100 apples demonstrated that the width of the apples was measured with a mean error of % 2.09, and apples were classified with a success rate of % 94.

Keywords: FPGA, VHDL, K-means, Apple Classification

1. Giriş

Endüstride birçok alanında nesne tanıma, sınıflandırma ve boyut ölçümü gibi işlemleri yapabilen otomasyon sistemlerine duyulan ihtiyaç gün geçtikçe artmaktadır. Bu sistemler yüksek performans ve hızla gıda sektöründe sınıflandırma amacıyla kullanılmaktadır. Bu otomasyon sistemleri genellikle masaüstü bilgisayar ya da dizüstü bilgisayarlara bağlı

olarak çalıştığından mobil değildirler. Bu durum sistemin taşınabilirliği açısından dezavantajlara yol açmaktadır. Bu çalışmada bu problemlere çözüm sunabilmek için mobil düzeyde çalışan, elma sınıflandırma ve genişlik ölçüm sisteminin tasarımı FPGA donanımıyla gerçekleştirilmiştir. FPGA, Xilinx firması tarafından 1980'li yıllarda tanıtılmıştır. Bu donanımların yüksek işlem performansı, geniş depolama özellikleri ve üstün

devre mimarileri gibi avantajları bulunmaktadır [1].

FPGA teknolojisindeki hızlı gelişmeler ve satış fiyatlarının düşmesi, bu donanımların görüntü işleme uygulamaları, nesne tanıma ve veri madenciliği gibi alanlarda kullanımının yaygınlaşmasını sağlamıştır. FPGA'larla tasarım işlemi gerçekleştirilirken gerekli değişiklikler kolayca yapılabilmektedir ve simülasyon yazılımı hatalı donanım tasarımlarının görülmesini sağlamaktadır. Bu sistemlerin en önemli avantajlarından biriside yüksek performans gerektiren durumlarda paralel işlem yapabilmeleridir. Bu özelliklerinden dolayı görüntü işleme, veri madenciliği, nesne tanıma gibi uygulamalarda kullanımları yaygınlaşmaya başlamıştır. FPGA'ların diğer avantajı da bilgisayardan bağımsız bir şekilde mobil işlem yapabilmeleri ve üretmiş oldukları sonuçları ise sisteme dâhil edilmiş görüntüleme donanımlarında yâda monitörlerde gösterebilmeleridir. Bu avantajlarından dolayı [2] çalışmamızda FPGA kullanımı tercih edilmiştir.

Bu çalışmada FPGA ile 50 adet starking(ST) ve 50 adet granny smith (GS) türü elmanın gerçek zamanlı olarak sınıflandırması ve genişliğinin ölçülmesi işlemleri gerçekleştirilmiştir. İşlem sonuçları ise FPGA kartı üzerindeki LED ve LCD ekranda gösterilmiştir. Mobil düzeyde çalışmak üzere tasarlanan sistemin yüksek hızla sınıflandırma işlemlerini yapabildiği görülmüştür.

Meyve kalite kontrol, sınıflandırma ve segmentasyon konusunda yapılan çalışmalarda [3-7] meyve sınıflandırma işleminin genellikle masaüstü yâda dizüstü bilgisayarlar üzerinde yapıldığı görülmüştür. Naik ve ark. [8] meyve sınıflandırma ve derecelendirmesinde kullanılan temel süreçleri incelemiştir. Bu kapsamda kullanılan özellik çıkarım ve makine öğrenmesi metotları gözden geçirilmiştir. Dara ve ark.[9] çalışmalarında elma meyvelerini boyut gruplarına (Küçük, Orta ve Büyük) ayırmaya yönelik görüntü işleme ve YSA teknikleri kullanarak bir sistem tasarımı geliştirmişler. Jamdar ve ark. [10] elma meyvesi üzerindeki üç yaygın hastalığın saptanması ve sınıflandırılması için çalışma yapmışlardır. İlgili çalışmada segmentasyon için K-means kümeleme algoritması kullanılmıştır. Literatür özeti verilen

bu çalışmalarda da sınıflandırma işlemleri gömülü sistem üzerinde değil PC' de gerçekleştirilmiştir.

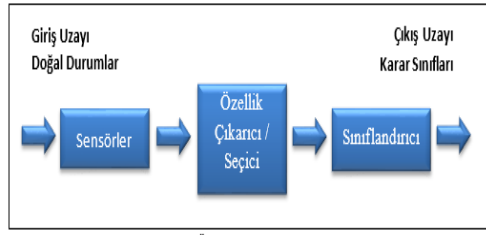
Pearson [11] çalışmasında renk farklılıkları veya küçük kusurları olan patlamış mısır ve buğday tanelerini tespit etmek ve ayırmak için FPGA tabanlı bir sistem tasarımı geliştirilmiştir. Nu'no-Maganda ve ark. [12] FPGA ile narenciye' nin piksel sınıflandırmasını yapmıştır. Ibarra-Manzano ve ark. [13] çalışmasında resimler üzerinde nesnelere saptamak için FPGA temelli mimariler önerilmiştir. Saegusa ve ark. [14] FPGA üzerinde çalışan K-means kümeleme temelli çalışan segmentasyon metodu sunmuşlardır.

Bu çalışmada elma ile zemini ayırmak ve sınıflandırma için K-means kümeleme algoritması kullanılmıştır. Ayrıca görüntüsü alınan her bir elmanın genişliğinin ölçümü de yapılmaktadır. Bu işlemlerin hepsi gerçek zamanlı olarak FPGA donanımı üzerinde yürütülmektedir. Fakat literatür çalışması yapıldığında meyve sınıflandırma sistemlerinin çoğunun gömülü sistemlerle değil PC tabanlı olarak çalıştığı görülmektedir. K-means algoritması danışmansız bir öğretim metodu olduğu için, yapay sinir ağları gibi zaman alıcı ve sorunların çıkabildiği eğitim sürecine gereksinim duymamaktadır. Bundan dolayı bu çalışmada K-means algoritması hızlı ve yüksek sınıflama doğruluğunda çözüm üretmiştir. Bu açıardan çalışma literatürdeki birçok meyve sınıflandırma sistemlerine göre üstünlükler içermektedir.

Çalışmada Bölüm 2'de örüntü tanıma, Bölüm 3'te sistemin çalışması, Bölüm 4'te deneysel sonuçlar ve Bölüm 5'te ise sonuç sunulmuştur.

2. Örüntü Tanıma

Ölçülendirme yapılabilen veya gözlenebilme özelliğinde olan bilgi türü örüntü olarak tanımlanmaktadır. Görüntü ve ses gibi bilgiler örüntü kavramı içerisine girmektedir. Örüntünün algılanması aşamasından sonra belirlenmiş olan kriterler doğrultusunda tanımlama ya da sınıflandırma işlemine örüntü tanıma denmektedir [15]. Örüntü tanıma blok diyagramı Şekil 1'de görülmektedir.

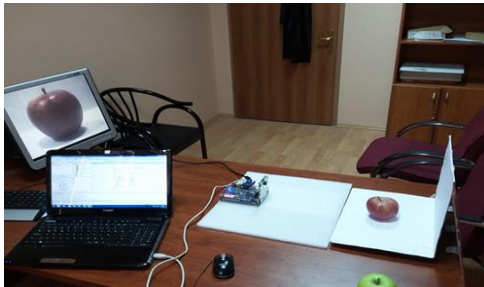


Şekil 1. Örüntü tanıma blok diyagramı

Algılayıcılar, herhangi bir anda dış ortamdan bilgi almaktadır. Burada kamera algılayıcı olarak kullanılmaktadır. Özellik çıkarıcı, yapılan ölçümlerin hepsinden oluşan giriş bilgilerinden sınıflandırıcı için gerekli olan bilgiyi çıkarmaktadır. Sınıflandırıcı ise belirlenen kısıtlara uygun olarak elmanın türünü tespit etmektedir. Algılayıcılar kısmında örüntünün (elma resmi) alınması gerçekleştirilmektedir. Özellik çıkarıcı/seçici bloğu sırasıyla, resmi gri düzeye çevirir ve filtrelemesini yapmaktadır. Sınıflandırıcı bloğu K-means kümeleme algoritması ile nesnenin sınıflandırılmasını yapmaktadır.

3. Sistemin Çalışması

Nesne tanıma ve genişlik ölçme sisteminde 100 adet elmanın genişlik ölçümü ve sınıflandırması gerçekleştirilmektedir. Şekil 2’de görülen elma sınıflandırma ve genişlik ölçüm sistemi için Altera DE-0 model FPGA kartı ve Terasic D5M kamera kullanılmıştır. 100 adet elma resminin aynı şekilde alınmasının ardından FPGA LED’lerinde sınıflandırma sonuçları gösterilmektedir.

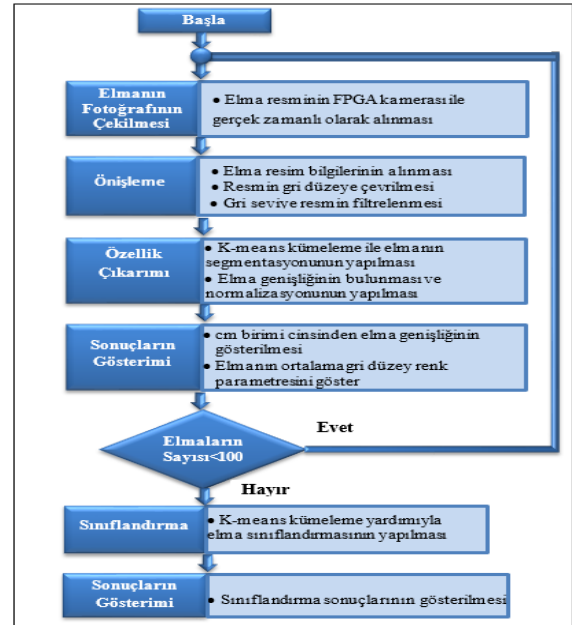


Şekil 2. Nesne Tanıma ve Genişlik Ölçüm Sisteminin Genel Yapısı

Şekil 3’te nesne tanıma ve genişlik ölçme sisteminin akış diyagramı görülmektedir. Sistemde ilk aşamada kameranın önüne

yerleştirilen elmanın fotoğrafı çekilmektedir ve ardından elmanın genişliği ve ortalama gri düzey değeri FPGA kartında gösterilecektir. İkinci adımda alınan resmin ön-işlemesi, üçüncü aşamada ön-işlemesi yapılan görüntünün özellik çıkarımı yapılmaktadır. Dördüncü adımda ise elmaya ait genişlik ve ortalama gri düzey değeri gösterilmektedir.

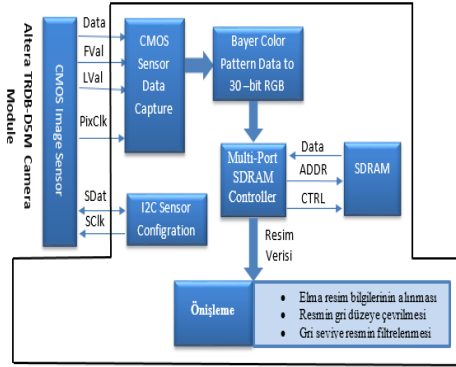
Bu işlem süreci 100 adet elma için tekrarlanacaktır. Son adımda 100 adet elmaya ait gri düzey verilerinin K-means kümeleme ile sınıflandırması gerçekleştirilerek elma sınıflandırma bilgileri FPGA kartı üzerindeki ledlerde gösterilecektir.



Şekil 3. Nesne tanıma ve ölçme sisteminin blok diyagramı

3. 1. Elmanın fotoğrafının çekilmesi

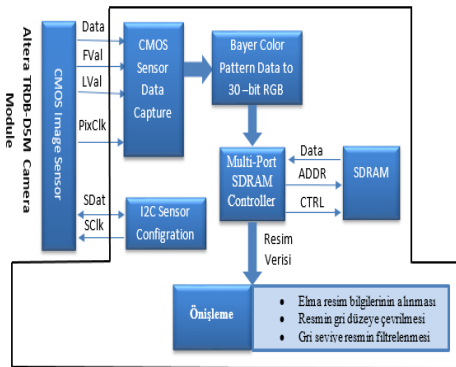
Bu bölümde elmanın resmi çekilip resim verileri ön-işleme bloğuna verilmektedir. Şekil 4’te bu süreçte yapılan işlemlerin akış diyagramı verilmiştir. FPGA kartı üzerinde bulunan butona basıldığında elma resmi çekilip RGB formatına çevrilmesinin ardından SDRAM’ e gönderilecektir. Sonra görüntü verisi SDRAM’ dan alınarak ön-işlemeye tabi tutulacaktır.



Şekil 4. Elmanın Fotoğrafının Alınmasını Gösteren Akış Diyagramı

3.1. Elmanın fotoğrafının çekilmesi

Bu bölümde elmanın resmi çekilip resim verileri önişleme bloğuna verilmektedir. Şekil 4'te bu süreçte yapılan işlemlerin akış diyagramı verilmiştir. FPGA kartı üzerinde bulunan butona basıldığında elma resmi çekilip RGB formatına çevrilmesinin ardından SDRAM'e gönderilecektir. Sonra görüntü verisi SDRAM'dan alınarak ön-ışlemeye tabi tutulacaktır.



Şekil 4. Elmanın Fotoğrafının Alınmasını Gösteren Akış Diyagramı

3.2. Önişleme

Başlangıçta elma resmine ait R(Red), G(Green), B(Blue) bilgileri denklem 1'deki formüle bağlı olarak işlenerek resmin gri formata [15] dönüştürülmesi sağlanmaktadır. Ardından gri düzeydeki resme 3x3'lük ortalama (average) filtresi [16,17] uygulanmaktadır. Ortalama filtresinin uygulanmasında amaç gri formattaki resmin pikselleri arasında yumuşak geçiş sağlanarak parazit ve gürültülerin giderilmesini

sağlamaktadır. Şekil 5'te önişleme adımının özeti verilmiştir.

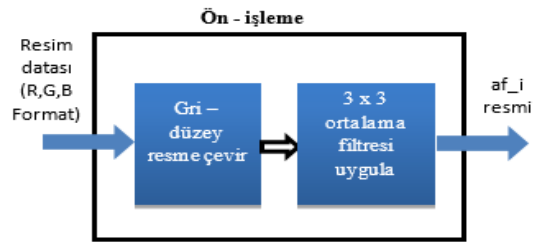
$$Grayscale = \frac{R+G+B}{3} \quad (1)$$

3.3. K-means kümeleme algoritması

K-means en yaygın kullanılan danışmansız öğrenme yöntemlerinden biridir [17]. Bu yöntem, tüm verilerin tek bir kümeye dâhil olmasını sağlar. Bu mekanizma N adet veriden oluşan bir veri dizisinin C kümeye bölünmesini sağlamaktadır. Bu işlem sürecinin en sonunda temel amaç, bölünme işlemi sonunda kümelerde yüksek düzeyde benzerlik ve kümeler arasında düşük benzerlik düzeyine sahip olmaktır. Kümelemede yaygın şekilde karesel hata kriteri (HK) kullanılmaktadır. HK'nın değeri düştükçe kümeleme başarımı artmaktadır. Verilerin buldukları kümenin merkez noktalarına olan uzaklıklarının karelerinin toplamı aşağıdaki denklem [17] aracılığıyla bulunmaktadır:

$$HK = \sum_{x=1}^C \sum_{y=1}^N (mesafe^2(a_x, b_y)) \quad (2)$$

burada N değeri dizideki toplam eleman sayısı, C küme sayısıdır. mesafe iki veri arasındaki Öklid uzaklığı, b değeri kümedeki bir eleman, a değeri küme merkezidir.



Şekil 5. Önişleme adımı akış diyagramı

3.4. Segmentasyon süreci

Bu adımda, K-means kümeleme algoritması uygulanarak elmanın segmentasyonu yapılmaktadır. Bu işlemi yapmak için hazırlanan kodlar Algoritma 1'de verilmiştir. Algoritma aşağıda sıralanan adımları içermektedir:

a. Başlangıç küme merkezlerinin belirlenmesi

Çalışmada küme sayısı 2 olarak ayarlandığından rastgele olarak seçilen 2 adet parlaklık değeri küme merkez başlangıç noktası olarak ayarlanmıştır. Algoritma 1'de 1 numaralı satırda bu işlem gerçekleştirilmektedir. Belirlenen iki farklı değer c_1 ve c_2 değişkenlerine aktarılmıştır.

b. Küme merkezleri ile olan mesafelerin hesaplanması

Bu aşamada af_i resminin her bir pikselinin parlaklık değeri ile c_1 ve c_2 arasındaki mesafe bulunmaktadır. Algoritma 1'de 8 ve 9 numaralı satırlarda bulunan mesafe fonksiyonu ile bu işlem gerçekleştirilmektedir. *mesafe* fonksiyonuna ait kodlar ise Algoritma 2'de verilmiştir.

Algoritma 1. Modifiye edilmiş k-means kümeleme algoritması.

```

1.  $(c_1, c_2) \leftarrow (\text{random}(0,255), \text{random}(0,255));$  -- rastgele seçilmiş küme merkezleri
2.  $devam \leftarrow 1;$ 
3. while( $devam = 1$ ) loop
4.    $(c1\_toplamlam, c2\_toplamlam, c1toplamlam, c2toplamlam) \leftarrow 0;$ 
5.    $karşılaştır\_kmean \leftarrow segmentasyon;$ 
6.   for  $i$  in  $0$  to  $height-1$  loop
7.     for  $i$  in  $0$  to  $width-1$  loop
8.        $mesafe1 \leftarrow mesafe(af\_i(i, j), c1);$ 
9.        $mesafe2 \leftarrow mesafe(af\_i(i, j), c2);$ 
10.      if  $mesafe1 < mesafe2$  then
11.         $segmentasyon(i, j) \leftarrow 0;$ 
12.         $c1\_toplamlam := af\_i(i, j) + c1\_toplamlam;$ 
13.         $c1toplamlam \leftarrow c1toplamlam + 1;$ 
14.      elsif  $mesafe1 > mesafe2$  then
15.         $segmentasyon(i, j) \leftarrow 1;$ 
16.         $c2\_toplamlam \leftarrow af\_i(i, j) + c2\_toplamlam;$ 
17.         $c2toplamlam \leftarrow c2toplamlam + 1;$ 
18.      end if;
19.    end loop;
20.  end loop;
21.  if ( $karşılaştır\_kmean = segmentasyon$ ) then
22.     $devam \leftarrow 0;$ 
23.  end if;
24.   $c1 \leftarrow c1\_toplamlam / c1toplamlam;$ 
25.   $c2 \leftarrow c2\_toplamlam / c2toplamlam;$ 
26. end loop;

```

c. Segmentasyonun yapılması ve kümelemelere ait veri toplamlarının bulunması

Bu adımda kümelere ait verilerin toplanması ve segmentasyon işlemi için gerekli hesaplamalar yapılmaktadır. Algoritma 1'de 10 ile 18. satırlar arasında bu süreç gerçekleştirilmektedir. Böylece segmentasyon değişkeninde segmente edilmiş resim tutulmaktadır. $c1_toplamlam$ 'da c_1 kümesi, $c2_toplamlam$ 'da c_2 kümesine ait verilerin toplamı, $c1toplamlam$ ve $c2toplamlam$ değişkenlerinde ise ilgili kümelerdeki eleman sayıları tutulmaktadır.

Algoritma 2. *mesafe* fonksiyonu.

```

1. function mesafe (constant  $A, B$ : real) return real is
2. begin
3.   return  $abs((A-B));$ 
4. end mesafe;

```

d. İterasyonun durdurulması

Bu adımla, en ideal segmentasyon resmi elde edildiğinde iterasyonun durdurulması sağlanmaktadır. Algoritma 1'de 5. satırda $karşılaştır_kmean$ değişkeni segmentasyon değişkenine eşitlenmektedir. segmentasyon işleminden sonra *segmentasyon* resmi ile $karşılaştır_kmean$ değişkenleri eşitse iterasyonun durdurulması sağlanmaktadır. Bu işlem Algoritma 1'de 21 ve 23. satırlar arasındaki kodlarla gerçekleştirilmektedir.

e. Küme merkezlerinin bulunması

$c1_toplamlam$, $c2_toplamlam$ değişkenleri, sırasıyla $c1toplamlam$, $c2toplamlam$ 'a bölünmesiyle yeni küme merkezleri olan c_1 , c_2 elde edilmektedir. Algoritma 1'de 24 ve 25. satırlarda bu işlem gerçekleştirilmektedir. c_1 'de elmaya ait küme merkezi bilgisi, c_2 'de ise arka plana ait küme merkezi bilgisi bulunmaktadır.

3.5. Elma genişliğinin ölçülmesi

Bu adımda piksel cinsinden elma genişliğini bulunmaktadır. Bu işlem için kullanılan kodlar, Algoritma 3'te verilmiştir. İlgili algoritmayla segmentasyon değişkeni kullanılarak en üst satırdan başlanarak en alt satıra kadar piksel

bazında resmin incelenmesi gerçekleştirilmektedir. İncelenen satırdaki elma genişliği satır_genişlik değişkenine

Algoritma 3. Elmanın genişliğini elde etmek için kullanılan algoritmik süreç.

1. $en_uzun_genişlik \leftarrow satır_genişlik$;
2. *for* i *in* 0 *to* $height-1$ *loop*
3. $satır_genişlik \leftarrow 0$;
4. *for* j *in* 0 *to* $width-1$ *loop*
5. *if* $segmentasyon(i, j)=0$ *then*
6. $satır_genişlik \leftarrow satır_genişlik+1$;
7. *end if*;
8. *end loop*;
9. *if* $satır_genişlik > en_uzun_genişlik$ *then*
10. $en_uzun_genişlik \leftarrow satır_genişlik$;
11. *end if*;
12. *end loop*

3.6. Sonuçların gösterilmesi

Önceki adımda elde edilen elma genişlik değeri piksel türünde olduğundan cm'ye çevrilmesi gerekmektedir. Önceden belirlenmiş olan kalibrasyon parametresi (KP) ile $en_uzun_genişlik$ bölünerek cm cinsinden elma genişliğinin bulunması sağlanmaktadır:

$$elma_genişliği = en_uzun_genişlik / KP \quad (3)$$

Şekil 6' da görülebildiği gibi, SW0 anahtarı 0 konumuna getirildiğinde FPGA kartındaki LCD ekranda elma genişliği değişken değeri, 1 konumuna getirildiğinde zaman c1 değeri gösterilmektedir. Toplam 100 adet elma için ayrıntısı yukarıda verilen segmentasyon süreci tekrarlanarak, bulunan c1 değerleri ile clstr dizisi elde edilmiştir

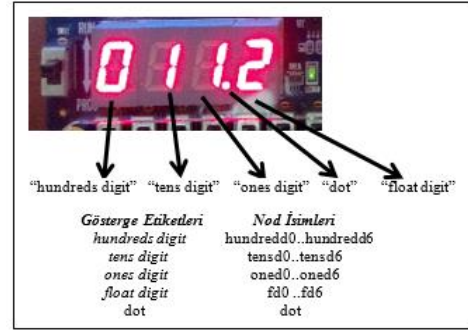


Şekil 6. elma_genişliği ve c1 değişkenlerinin gösterilmesi

aktarılmaktadır. Elmanın en uzun genişlik değeri ise $en_uzun_genişlik$ değişkenine aktarılmıştır.

3.7. Göstergelerin adreslenmesi

FPGA kartı üzerindeki LCD elemanı Şekil 7' de gösterildiği gibi, "hundreds digit", "tens digit", "dot", "ones digit" ve "float digit" olarak isimlendirilmiştir. Bu gösterge isimlerine karşılık gelen nod isimleri ve adreslemeleri Şekil 8' de verilmiştir.



Şekil. 7 Gösterge etiketleri

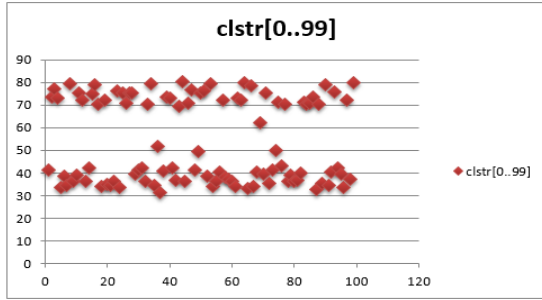
Node Name	PCB layer	I/O Bank	Location	Direction	VREF Group
dot	7	PIN_B15	PIN_B15	Output	B7_N1
fd0	7	PIN_E11	PIN_E11	Output	B7_N1
fd1	7	PIN_F11	PIN_F11	Output	B7_N1
fd2	7	PIN_H12	PIN_H12	Output	B7_N1
fd3	7	PIN_H13	PIN_H13	Output	B7_N1
fd4	7	PIN_G12	PIN_G12	Output	B7_N1
fd5	7	PIN_F12	PIN_F12	Output	B7_N1
fd6	7	PIN_F13	PIN_F13	Output	B7_N1
hundredd0	7	PIN_B18	PIN_B18	Output	B7_N0
hundredd1	7	PIN_F15	PIN_F15	Output	B7_N0
hundredd2	7	PIN_A19	PIN_A19	Output	B7_N0
hundredd3	7	PIN_B19	PIN_B19	Output	B7_N0
hundredd4	7	PIN_C19	PIN_C19	Output	B7_N0
hundredd5	7	PIN_D19	PIN_D19	Output	B7_N0
hundredd6	7	PIN_G15	PIN_G15	Output	B7_N0
oned0	7	PIN_A13	PIN_A13	Output	B7_N1
oned1	7	PIN_B13	PIN_B13	Output	B7_N1
oned2	7	PIN_C13	PIN_C13	Output	B7_N1
oned3	7	PIN_A14	PIN_A14	Output	B7_N1
oned4	7	PIN_B14	PIN_B14	Output	B7_N1
oned5	7	PIN_E14	PIN_E14	Output	B7_N1
oned6	7	PIN_A15	PIN_A15	Output	B7_N1
reset1	7	PIN_A18	PIN_A18	Output	B7_N0
reset2	7	PIN_G16	PIN_G16	Output	B7_N0
reset3	7	PIN_D13	PIN_D13	Output	B7_N1
tensd0	7	PIN_D15	PIN_D15	Output	B7_N0
tensd1	7	PIN_A16	PIN_A16	Output	B7_N1
tensd2	7	PIN_B16	PIN_B16	Output	B7_N1
tensd3	7	PIN_E15	PIN_E15	Output	B7_N0
tensd4	7	PIN_A17	PIN_A17	Output	B7_N1
tensd5	7	PIN_B17	PIN_B17	Output	B7_N1
tensd6	7	PIN_F14	PIN_F14	Output	B7_N0

Şekil. 8 Gösterge nodlarının adreslenmesi

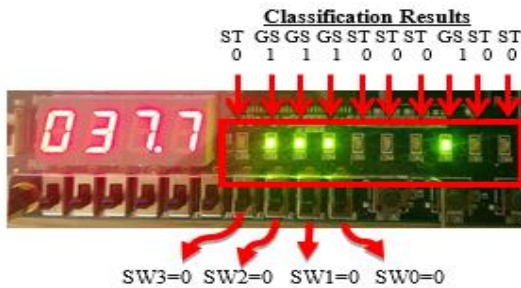
3.8. Sınıflandırma

Bu aşamada, elmaların sınıflandırması gerçekleştirilmektedir. Sınıflandırma işleminin başlangıcında küme sayısı 2'ye ayarlanmıştır. Bu işlemin ardından clstr dizisindeki verilerin K-means kümeleme algoritmasına bağlı olarak sınıflandırması gerçekleştirilmektedir. ST elmaya ait c1 düzeyi 33.8- 42.2 arasında olup, GS elmanın 70.1-79.4 arasındadır. clstr dizisinin grafiksel gösterimi Şekil 9' da verilmiştir.

Hazırlanan yazılımda clstr dizisinin K-means kümeleme yöntemi ile sınıflandırma sonucu clusteroutt değişkeninde tutulmuştur. Bu sınıflandırılma sonucunda, ST tür elmaların c1 değeri GS elmanınkinden daha düşük olduğu için GS tür elmalar 0 değeriyle, ST elmalar ise 1 ile temsil edilmektedir. Şekil 10' de görülen FPGA kartı üzerinde bulunan SW0, SW1, SW2 ve SW3 anahtarının pozisyonlarına bağlı olarak onlu gruplar halinde bu sınıflandırma sonuçları LED'lerde gösterilmektedir. clusteroutt için adresleme haritası Şekil 11' de verilmiştir. 100 adet elmada elde edilen clstr ve clusteroutt değerleriyle clstr ve clusteroutt dizileri elde edilmiştir.



Şekil. 9 K-means kümeleme sonuçları



Şekil. 10 FPGA kartı üzerinde sınıflandırma sonuçlarının gösterilmesi

Node Name	Direction	Location	I/O Bank	REF Group
clusteroutt[9]	Output	PIN_B1	1	B1_NO
clusteroutt[8]	Output	PIN_B2	1	B1_NO
clusteroutt[7]	Output	PIN_C2	1	B1_NO
clusteroutt[6]	Output	PIN_C1	1	B1_NO
clusteroutt[5]	Output	PIN_E1	1	B1_NO
clusteroutt[4]	Output	PIN_F2	1	B1_NO
clusteroutt[3]	Output	PIN_H1	1	B1_N1
clusteroutt[2]	Output	PIN_J3	1	B1_N1
clusteroutt[1]	Output	PIN_J2	1	B1_N1
clusteroutt[0]	Output	PIN_J1	1	B1_N1

Şekil. 11 clusteroutt için adresleme haritası

4. Deneysel Sonuçlar

Tasarlanan nesne tanıma ve ölçüm sisteminin performansını incelemek için farklı ölçülere sahip 50 adet ST, 50 adet GS tür elmalarla ait özellik çıkarım analizi ve sınıflandırma testi yapılmıştır.

4.1. Özellik çıkarımı analizi

Tasarlanan sistemin elma genişliği ölçüm başarımı bu test sürecinde gerçekleştirilmiştir. Bu test sürecinde aşağıda verilen denkleme bağlı olarak sistemin tespit ettiği ölçülen elma genişliği ve gerçek elma genişliği karşılaştırılarak mutlak hata bulunmaktadır:

$$\Delta m (\text{mutlak hata}) = |(\text{ölçülen elma_genişliği}) - (\text{gerçek elma_genişliği})| \quad (4)$$

İlk test işleminde Şekil 12.a'da görülen gerçek elma genişliği 7.42 cm olan ST tür elma kullanılmıştır. Test sonucunda sw0 anahtarı 0'a ayarlanarak ölçülen elma genişliği değeri 7.6 cm ve sw0 anahtarı 1'e ayarlanarak c1 değeri 41.3 olarak elde edilmiştir. Şekil 12' de bu sürecin görseli sunulmuştur. Δm değeri ise aşağıdaki şekilde hesaplanmıştır:

$$\Delta m = |7.6 - 7.42| = 0.18 \text{ cm} \quad (5)$$

Yukarıda anlatılan test işlemlerine benzer şekilde 99 adet elma (GS ve ST türünden) üzerinde de benzer analizler gerçekleştirilmiştir. Test işlemi sonucunda oluşan Δm sonuçlarının istatistiksel analizi gerçekleştirilmiş ve sonuçlar Tablo 1' de sunulmuştur. İlgili tablodan 100 test' den elde edilen Δm sonuçlarının aritmetik ortalaması 0,209 olarak elde edilmiştir. Bu durum yüksek doğrulukta elma genişliğinin ölçüldüğünü göstermektedir. Deney numarasına

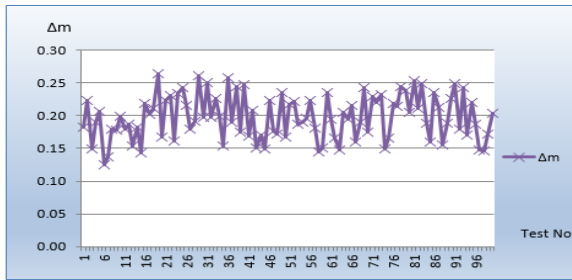
bağlı olarak elmaların Δm değişimi grafiği Şekil 13’ de verilmiştir. İlgili grafikten Δm sonuçlarının oldukça düşük düzeyde olduğu görülmektedir.



(a)

(b)
sw0=0(c)
sw0=1

Şekil 12. Analiz yapılan elma ve test sonuçları

Şekil 13. Elma genişlik ve Δm değişim grafiği

Tablo 1. GS tür elma analizinin istatistiksel analiz sonuçları

Δm için istatistiksel sonuçlar		
Maksimum	Aritmetik ortalama	Standart sapma
0,3	0,209	0,0413

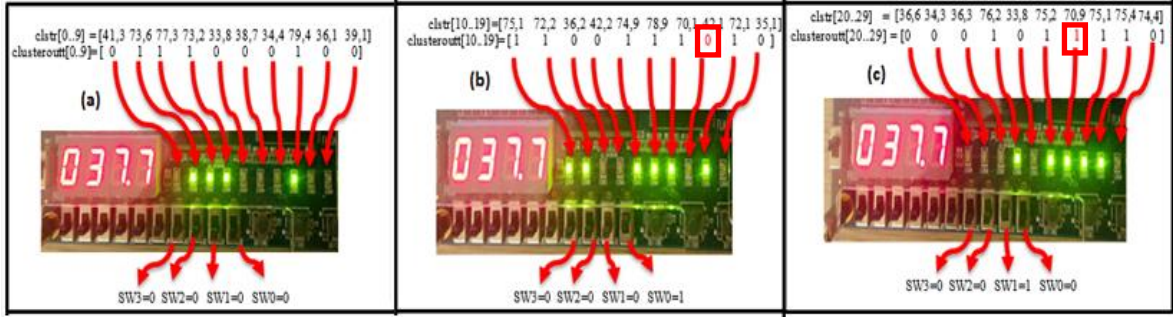
4.2. Sınıflandırma analizi

Şekil 14. a’ da görüldüğü gibi, FPGA kartı üzerinde bulunan anahtarlar SW0 = 0, SW1 = 0, SW2 = 0 ve SW3 = 0 kombinasyonunda iken, ilk on elmanın sınıflandırma sonuçlarına bağlı olarak (clusteroutt[0..9]) FPGA kartındaki LED’lerde yanmaktadır. Bu sınıflandırma sonuçlarında hata olmadığından 100%’ lük bir başarı elde edilmiştir.

Şekil 14. b’ de görüldüğü gibi, FPGA kartı üzerinde bulunan anahtarlar SW0 = 0, SW1 = 0, SW2 = 0 ve SW3 = 1 kombinasyonunda iken, ikinci 10’ lu grup elmaların sınıflandırma sonuçlarına (clusteroutt[11..19]) bağımlı olarak FPGA kartındaki ledler yanmaktadır. Bu sınıflandırma sonuçlarında 1 adet hatalı sonuç oluştuğundan 90%’ lık bir başarı elde edilmiştir.

Şekil 14. c’ de görüldüğü gibi, FPGA kartı üzerinde bulunan anahtarlar SW0 = 0, SW1 = 0, SW2 = 1 ve SW3 = 0 kombinasyonunda iken, üçüncü 10’ lu grup elmaların sınıflandırma sonuçlarına (clusteroutt[21..29]) bağımlı olarak FPGA kartındaki ledler yanmaktadır. Bu sınıflandırma sonuçlarında da 1 adet hatalı sonuç oluştuğundan 90%’ lık bir başarı elde edilmiştir.

Yukarıda ayrıntılı olarak sunulan 30 elma testi sonucu Tablo 2’ de Test no. 1 ve Test no. 3 arasında sunulmuştur. Bu test işlemlerine ek olarak geriye kalan 70 adet elma için, SW0, SW1, SW2 ve SW3 anahtarlarının pozisyonlarına bağlı olarak elde edilen sınıflandırma sonuçları Test no. 31 ve Test no. 100 arasında sunulmuştur. Tabloda hatalı sınıflandırma sonuçları kırmızı renkli, altı çizgili ve italik olarak gösterilmiştir. Bu testlerin sonucunda, toplam 100 elma sınıflandırmasında % 94’ lük başarı oranı elde edilmiştir.



Şekil 14. SW3, SW2, SW1 ve SW0' ın pozisyonuna bağlı olarak sınıflandırma sonuçlarının gösterilmesi

Tablo 2. Test sonuçları

Anahtarların konumları ve Test Numaraları	Results
<i>Test no 1-10</i>	<i>clstr[0..9] değerleri ve clusterout[0..9] sonuçları</i>
SW0=0 SW1=0 SW2=0 SW3=0	41.3 73.6 77.3 73.2 33.8 38.7 34.4 79.4 36.1 39.1 0 1 1 1 0 0 0 1 0 0
<i>Test no 11-20</i>	<i>clstr[10..19] değerleri ve clusterout[10..19] sonuçları</i>
SW0=0 SW1=0 SW2=0 SW3=1	75.1 72.2 36.2 42.2 74.9 78.9 70.1 <u>42.1</u> 72.1 35.1 1 1 0 0 1 1 1 <u>0</u> 1 0
<i>Test no 21-30</i>	<i>clstr[20..29] değerleri ve clusterout[20..29] sonuçları</i>
SW0=0 SW1=0 SW2=1 SW3=0	36.6 34.3 36.3 76.2 33.8 75.2 <u>70.9</u> 75.1 75.4 74.4 0 0 0 1 0 1 <u>1</u> 1 1 1
<i>Test no 31-40</i>	<i>clstr[30..39] değerleri ve clusterout[30..39] sonuçları</i>
SW0=0 SW1=0 SW2=1 SW3=1	71.4 42.1 36.5 <u>61.4</u> 79.2 34.5 40.6 35.4 40.9 73.4 1 0 0 <u>1</u> 1 0 0 0 0 1
<i>Test no 41-50</i>	<i>clstr[40..49] değerleri ve clusterout[40..49] sonuçları</i>
SW0=0 SW1=1 SW2=0 SW3=0	73.2 42.1 36.9 76.3 70.4 36.3 <u>65.6</u> 76.5 71.5 70.3 1 0 0 1 1 0 <u>1</u> 1 1 1
<i>Test no 51-60</i>	<i>clstr[50..59] değerleri ve clusterout[50..59] sonuçları</i>
SW0=0 SW1=1 SW2=0 SW3=1	75.2 76.4 38.7 78.6 33.9 36.3 40.5 72.1 38.3 37.3 1 1 0 1 0 0 0 1 0 0
<i>Test no 61-70</i>	<i>clstr[60..69] değerleri ve clusterout[60..69] sonuçları</i>
SW0=0 SW1=1 SW2=1 SW3=0	36.2 34.1 73.2 72.1 73.8 71.1 78.5 34.0 70.2 <u>71.5</u> 0 0 1 1 1 0 1 0 1 <u>1</u>
<i>Test no 71-80</i>	<i>clstr[70..79] değerleri ve clusterout[70..79] sonuçları</i>
SW0=0 SW1=1 SW2=1 SW3=1	39.6 75.2 35.2 41.2 <u>46.9</u> 71.4 40.2 70.1 36.1 39.1 0 1 0 0 <u>0</u> 1 0 1 0 0
<i>Test no 81-90</i>	<i>clstr[80..89] değerleri ve clusterout[80..89] sonuçları</i>
SW0=1 SW1=0 SW2=0 SW3=0	36.4 36.7 40.1 71.4 70.5 73.6 73.7 36.5 70.4 35.4 0 0 0 1 1 1 1 0 1 0
<i>Test no 91-100</i>	<i>clstr[90..99] değerleri ve clusterout[90..99] sonuçları</i>
SW0=1 SW1=0 SW2=0 SW3=1	79.1 74.6 40.3 75.6 42.1 39.6 37.7 72.1 77.3 76.1 1 1 0 1 0 0 0 1 1 1

5. Sonuçlar

Bu çalışmada gıda sektöründe ihtiyaç duyulan bir sınıflandırma sisteminin mobil düzeyde tasarımının yapılması hedeflenmiştir. Bu kapsamda, GS ve ST tür elmaların genişliklerini ölçen ve renge bağlı olarak elmaları sınıflandıran mobil bir FPGA donanımı geliştirilmiştir. Başlangıçta kameradan gerçek zamanlı olarak görüntü alınmasının ardından bu görüntü gri düzeye çevrilmiştir. Ardından k-means kümeleme yöntemiyle görüntünün segmentasyonu yapılmıştır. Çalışmamızda segmentasyon başarısını arttırmak için K-means metodu modifiyeler içerecek şekilde VHDL dilinde yazılmıştır. Bu işlemin ardından K-means kümeleme yöntemiyle elmaların sınıflandırması yapılmıştır. Kümeleme sonucu LED'lerde ve elma genişlik değeri ise FPGA göstergesinde gösterilmiştir. Yapılan test çalışmaları sonucunda % 94 başarı oranı ile elmaların sınıflandırıldığı ve düşük mutlak hatayla elma genişliğinin ölçüldüğünü göstermiştir.

6. Kaynaklar

1. Volnei A. (2004). Pedroni, Circuit Design with VHDL. Massachusetts Institute of Technology, London, England.
2. Hauck, S., DeHon, A. (2008). Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation. Elsevier Inc, Burlington.
3. Rochaa, A., Hauaggeb, D. C., Wainera, J., Goldensteina, S. (2010). Automatic fruit and vegetable classification from images. *Computers and Electronics in Agriculture*, **70(1)**: 96–104.
4. Baranowski, P., Mazurek, W., Pastuszka-Woźniak, J. (2013). Supervised classification of bruised apples with respect to the time after bruising on the basis of hyperspectral imaging data. *Postharvest Biology and Technology*, **86**: 249–258.
5. Mizushima, A., Lub, R. (2013). An image segmentation method for apple sorting and grading using support vector machine and Otsu's method. *Computers and Electronics in Agriculture*, **94**: 29–37.
6. Unay, D., Gosselin, B. (2007). Stem and calyx recognition on 'Jonagold' apples by pattern recognition. *Journal of Food Engineering*, **78(2)**:597 – 605.
7. Dubey, S. R., Jalal, A. S. (2016). Apple disease classification using color, texture and shape features from images. *Signal, Image and Video Processing*, **10(5)**: 819–826, 2016
8. Naik, S., Patel, B. (2017). Machine vision based fruit classification and grading - a review. *International Journal of Computer Applications*, **170(9)**: 1-13.
9. Dara, F., Devolli, A. (2016). Applying artificial neural networks (ANN) techniques to automated visual apple sorting. *Journal of Hygienic Engineering and Design*, **17**: 55-63.
10. Jamdar, A. V., Patil, A. P. (2017). Detection and classification of apple fruit diseases using K-means clustering and learning vector quantization neural network. *IJSDR*, **2(6)**: 423-429.
11. Pearson, T. (2009). Hardware-based image processing for high-speed inspection of grains. *Computers and Electronics in Agriculture*, **69**: 12–18
12. Nuño-Maganda, M. A., Hernandez-Mier, Y., Torres-Huitzil, C., Jim'enez-Arteaga, J. (2014). FPGA-based real-time citrus classification system. *2014 IEEE 5th Latin American Symposium*, 25-28 Feb. 2014, Santiago, Chile.
13. Ibarra-Manzano, M.-A., Devy, M., Boizard, J.-L. (2010). Real-Time classification based on color and texture attributes on an FPGA-based architecture. *Design and Architectures for Signal and Image Processing (DASIP)*, 26-28 October 2010, Edinburgh, UK.
14. Saegusa, T., Maruyama, T. (2007). Real-time segmentation of color images based on the K-Means clustering on FPGA, Field-Programmable Technology. *ICFPT 2007 International Conference*, 12-14 December 2007, Kitakyushu, Japan.
15. Avci, E., Turkoglu, İ., Poyraz, M. (2005). Intelligent target recognition based on wavelet packet neural network, *expert systems with applications*, **29**: 175–182.
16. Rafael, C. G., Woods, R. E. (2002). Digital Image Processing. *Prentice Hall, U.S.A.*
17. Sert, E. and Okumuş, İ. T. (2014). Segmentation of mushroom and cap width measurement using modified K-means clustering algorithm. *Advances in Electrical and Electronic Engineering Journal*, **12(4)**: 354-360. 2014.