Dikkat Katmanlı Derin Sinir Ağları ile Anahtar Kare Tespiti

Key Frame Extraction with Attention Based Deep Neural Networks

Samed Arslan[1], *Senem Tanberk*[2]

[1]Research and Innovation
Huawei Turkey Research and Development Center, Istanbul
samedarslan90@gmail.com

[2]Research and Innovation
Huawei Turkey Research and Development Center, Istanbul
0000-0003-1668-0365

Özetçe

Anahtar kare tespiti, uzun videoları en iyi özetleyebilecek sahnelerin seçilmesi şeklinde yapılan bir çalışmadır. Videonun özetini çıkarmak, hızlı göz atmayı ve içerik özetlemeyi kolaylaştırmak veya bazı görüntü işleme yöntemlerinde ön işleme çalışmalarında kullanmak için önemli bir görevdir. Elde edilen fotoğraflar farklı sektörlerde otomatikleştirilmiş işler (örneğin; güvenlik görüntülerinin özetlenmesi, müzik kliplerinde kullanılan farklı sahnelerin tespit edilmesi) için kullanılmaktadır. Bunun dışında ileri makine öğrenim yöntemlerinde yüksek hacimli videoların işlenmesi kaynak maliyeti yaratır. Elde edilen anahtar kareler; kullanılacak yöntemlere ve modellere girdi özellik olarak kullanılabilmektedir. Bu çalışmada; dikkat katmanına sahip derin bir oto kodlayıcı model kullanarak, anahtar kare tespiti için, derin öğrenmeye dayalı bir yaklaşım önerilmektedir. Önerilen yöntemde, önce otomatik kodlayıcının kodlayıcı kısmını kullanılarak video karelerinden öznitelikleri çıkarılır ve bu nitelikler ile benzer kareleri bir arada gruplamak için k-ortalama kümeleme algoritması kullanılarak segmentasyon uygulanır. Ardından, kümelerin merkezine en yakın kareler belirlenir her kümeden anahtar kareler seçilir. Metot TVSUM video veri setinde değerlendirilmiştir ve 0,77'lik bir sınıflandırma doğruluğu elde edilmiştir, bu da birçok mevcut metottan daha yüksek bir başarım oranını göstermektedir. Önerilen yöntem, video analizinde anahtar kare çıkarma için umut verici bir çözüm sunmaktadır ve video özetleme ve video alma gibi çeşitli uygulamalara uygulanabilir.

Abstract

Keyframe detection is the study of selecting scenes that can best summarize long videos. Extracting a video summary is an important task to facilitate quick browsing and summarizing content, or for use in pre-processing in some image processing methods. The resulting photos are used for automated tasks in different industries (eg summarizing security footage, identifying different scenes used in music clips). In addition, processing high-volume videos in advanced machine learning methods creates resource costs. Keyframes obtained; It can be used as an input feature to the methods and models to be used. In this study; A deep learning-based approach is proposed for keyframe detection using a deep auto-encoder model with an attention layer. In the proposed method, first the features are extracted from the video frames using the encoder part of the autoencoder, and segmentation is applied using the k-means clustering algorithm to group similar frames together with these features. Then, the squares closest to the center of the clusters are determined and keyframes are selected from each cluster. The method was evaluated on the TVSUM video dataset and achieved a classification accuracy of 0.77, indicating a higher success rate than many existing methods. The proposed method offers a promising solution for keyframe extraction in video analysis and can be applied to a variety of applications such as video summarization and video retrieval.

1. Giriş

Video analizi, gözetim, eğlence ve eğitim gibi çok çeşitli alanlardaki uygulamalarla, son yıllarda giderek daha önemli bir araştırma alanı haline gelmiştir. Anahtar kare çıkarma, video analizinde önemli bir çalışma alanıdır, çünkü önemli olayları veya eylemleri yakalayarak en bilgilendirici karelerin seçilmesi ile bir videonun içeriğinin özetlenmesine yardımcı olur [1]. Diğer bir deyişle, video anahtar kare çıkarmanın temel amacı, video içeriğini temsil edecek video karelerini tespit edilmesidir.

Tespit edilecek özet karelerin temsil özellikleri çalışma alanlarına göre farklılık gösterebilmektedir. Ancak en genel kullanım senaryosu, video akışı içerisindeki farklı sahne ve çevre unsurlarının tespit edilmesi veya genel akıştaki nesne hareket hızlarından daha farklı hızda hareket eden nesnelerin tespit edilebildiği karelerin bulunmasıdır.

Farklı veri setleri üzerinde yapılan başarılı çalışmalarda genel başarım %70 ile %90 civarında sınıflandırma başarımı göstermektedir [2]. Ancak bu durum veri setlerine göre değişiklik göstermektedir. Bazı veri setlerinde bu başarım daha aşağıda kalırken bazılarında daha yüksek olabilmektedir.

Geleneksel anahtar kare çıkartmaya yönelik yöntemler genellikle, buluşsal yöntemlere dayanmaktadır. Veya belirli video içeriğine büyük ölçüde bağlı olabilen ve diğer videolarda kullanılmak üzere genellenemeyen özel uygulamalardan oluşmaktadır. Bu çalışmada, dikkat

katmanına sahip derin bir otomatik kodlayıcı kullanarak anahtar kare çıkarma için yeni bir yaklaşım önerilmektedir. Kullanılacak modelin, anahtar kare çıkarma için ihtiyaç duyulan video içeriğinin özelliklerini otomatik olarak öğrenebilen bir yapıda olması planlanmaktadır [3].

Önerilen yöntemde, önce otomatik kodlayıcının kodlayıcı kısmı ve dikkat katmanı kullanılarak video karelerinden öznitelikler çıkarılır ve daha sonra bu öznitelikler K-means kümelemesi kullanılarak kümelenir. Küme merkezine en yakın olan video görüntüleri anahtar kare olarak seçilir. Otomatik kodlayıcının kodlayıcı kısmında dikkat katmanının kullanılması, video karelerindeki en belirgin özelliklerin vurgulanmasına yardımcı olur ve bu, anahtar kare çıkarma işleminin doğruluğunu artırır [4, 5].

Önerilen yöntemin etkinliğini değerlendirmek için, anahtar kare çıkarımı için bir kıyaslama veri kümesi üzerinde deneyler yapıldı. Deneysel sonuçlar, önerilen yöntemin, anahtar kare çıkarma için mevcut yöntemlerle karşılaştırılabilir olduğunu ve iyi performans gösterdiğini, mevcut yöntemlere alternatif olarak kullanılabileceğini, video özetleme ve eylem tanıma gibi çeşitli alanlarda potansiyel uygulamalara sahip olduğunu göstermektedir.

Bu çalışmanın geri kalanı şu şekilde düzenlenmiştir: Bölüm iki, dikkat katmanına sahip derin bir otomatik kodlayıcı kullanarak anahtar kare çıkarma için önerilen yöntemin ayrıntılı bir açıklamasını içerir. Veri seti hakkında bilgi verir. Bölüm üç, deneysel kurulumu ve değerlendirme sonuçlarını içerir. Son bölümde ise çalışma sonuçları değerlendirilir ve yorumlanır.

2. Yöntem

2.1. Ön işleme

Girdi olarak kullanılmak üzere, görüntü verisini düzenlemek gerekecektir. Bu adımlarda hem standardizasyon hem de normalizasyon çalışması yapılacaktır. Her adımın çalışma dökümü aşağıdaki gibidir:

İlk olarak, görüntünün renk uzayını BGR'den HSV'ye dönüştürülmesidir. Bu, renk bilgisini parlaklık bilgisinden ayırmak için bilgisayar görüşünde kullanılan yaygın bir tekniktir [6].

İkincisi, görüntüyü 64 x 64 piksel boyutlarında bir kare şeklinde yeniden boyutlandırılır. Görüntüyü daha küçük bir boyuta yeniden boyutlandırmak, görüntünün en önemli görsel özelliklerini korurken model üzerindeki hesaplama yükünün azaltılmasına yardımcı olabilir. Bu adımda yeterli kaynak sağlandığı durumlarda, daha büyük çözünürlükler genellikle daha başarılı sonuçlar üretmektedir.

Üçüncü olarak, görüntünün piksel değerleri 0 ila 255 aralığında normalleştirilir. Bu, tüm piksel değerlerinin tutarlı bir aralıkta olmasını sağlamak için görüntü işlemede kullanılan yaygın bir tekniktir.

Son olarak, önceden işlenmiş görüntü bir listeye eklenir. Kullanılmaya hazır hale getirilir.

Kısaca, bu adımlar bir görüntü almak, renk alanını HSV'ye dönüştürmek, daha küçük bir boyuta yeniden boyutlandırmak, piksel değerlerini normalleştirmek ve onu bir makine öğrenimi modeline beslenebilecek önceden işlenmiş kareler listesine eklemek şeklindedir.

2.2. Metotlar

2.2.1. Derin öğrenme ve otokodlayıcılar

"Derin öğrenme", çok katmanlı sinir ağlarına dayanan makine öğrenimi algoritmalarının bir alt kümesini ifade etmektedir. Bu algoritmalar, doğrusal olmayan dönüşümlerin birden çok katmanı aracılığıyla girdileri yinelemeli olarak dönüştürerek verilerin karmaşık temsillerini öğrenebilir [7].

"Otokodlayıcılar", veri sıkıştırma ve özellik çıkarma gibi denetimsiz öğrenme görevleri için derin öğrenmede kullanılan bir tür sinir ağı mimarisidir. Otomatik kodlayıcılar, orijinal girdi ile yeniden oluşturulmuş çıktı arasındaki farkı en aza indirerek girdi verilerini yeniden yapılandırmak için eğitilmiş bir kodlayıcı ve bir kod çözücü ağından oluşur [8].

Kodlayıcı ağı bir girdi alır ve onu, girdinin en önemli özelliklerini yakalayan daha düşük boyutlu bir temsile veya gizli alana eşler. Kod çözücü ağı, daha sonra gizli temsili alır ve onu orijinal girdi alanına geri eşler [9].

Otomatik kodlayıcılar, göreve bağlı olarak farklı kayıp işlevleriyle eğitilebilir. Örneğin, veri sıkıştırmada, kayıp fonksiyonu, orijinal girdi ile yeniden oluşturulmuş çıktı arasındaki ortalama karesel hata olabilir. Özellik çıkarmada, kayıp fonksiyonu, kodlayıcı ağının girişi ve çıkışı arasındaki fark olabilir [10].

2.2.2. Derin öğrenme dikkat katmanları (Attention layer)

Dikkat katmanları, görüntü ve video oluşturma, metin özetleme ve makine çevirisi gibi görevlerde performanslarını artırmak için otomatik kodlayıcılara eklenebilen bir tür mekanizmadır [11].

Dikkat katmanları, modelin çıktıyla alaka düzeyine bağlı olarak girdi verilerinin farklı bölümlerine seçici olarak odaklanmasını sağlar. Bu, çıktı için önemine bağlı olarak her girdi öğesine bir ağırlık atayarak ve bu ağırlıkları girdi öğelerinin ağırlıklı bir toplamını hesaplamak için kullanarak yapılır [12].

Otomatik kodlayıcılarda, girdideki önemli özellikleri seçerek, vurgulayarak yeniden oluşturulan çıktının kalitesini artırmak için dikkat katmanları kullanılabilir. Örneğin, görüntü oluşturmada, istenen çıktıya bağlı olarak girdi görüntüsünün farklı bölgelerine seçici olarak odaklanmak için dikkat katmanları kullanılabilir. Metin özetlemede, özetin uzunluğuna ve içeriğine bağlı olarak girdi belgesindeki önemli cümlelere veya anahtar sözcüklere seçici olarak odaklanmak için dikkat katmanları kullanılabilir[13, 14, 15].

Bu çalışmada dikkat katmanının video içeriğindeki ana özelliklere odaklanacağı varsayılmıştır. Bu özellikleri, modelin öğrenim aşamasında daha vurgulu hale getirerek anahtar karelerin daha başarımlı tespit edeceği öngörülmüştür.

2.2.3. Kümeleme

Kümeleme algoritmaları, veri noktalarını benzerliklerine göre gruplandırmak için kullanılan gözetimsiz makine öğrenimi algoritmalarının bir sınıfıdır. Kümelemenin amacı; grup etiketleri hakkında herhangi bir ön bilgi olmaksızın, verilerde doğal gruplamalar veya kümeler bulmaktır [16].

K-means gibi bölüm tabanlı kümeleme algoritmaları, noktalar ile bir küme merkezleri kümesi arasındaki mesafelere bağlı olarak veri noktalarını sabit sayıda kümeye böler. Merkezler, yakınsamaya ulaşılana kadar yinelemeli olarak güncellenir ve bu da verilerin kümelere bölünmesiyle sonuçlanır [17,18].

Görüntü analizi, metin madenciliği ve sosyal ağ analizi gibi çeşitli alanlardaki verileri analiz etmek için kümeleme algoritmaları kullanılabilir. Araştırmacılar, farklı türde kümeleme algoritmalarını keşfedebilir ve siluet katsayısı, saflık veya entropi gibi metrikleri kullanarak bunların etkililiğini değerlendirebilir. Girdi verileri, mesafe veya benzerlik ölçüsü ve algoritmanın performansını optimize etmek için kullanılan herhangi bir hiper parametre veya ayarlama yöntemi dahil olmak üzere, kullanılan kümeleme algoritmasının net bir tanımını sağlamak önemlidir [19,20].

2.2.4. Uzaklık metrikleri

Mesafe ölçümleri, iki veri noktası arasındaki benzerliği veya farklılığı ölçmek için kullanılan matematiksel işlevlerdir. Makine öğreniminde, mesafe ölçümleri genellikle özellik vektörleri veya veri noktaları arasındaki mesafeyi ölçmek için kullanılır ve genellikle kümeleme, sınıflandırma ve regresyon görevlerinde kullanılır [21,22].

Öklid mesafesi belki de en iyi bilinen mesafe metriğidir ve iki veri noktasında karşılık gelen özellik değerleri arasındaki farkların karelerinin toplamının karekökü olarak hesaplanır. Taksi mesafesi olarak da bilinen Manhattan mesafesi, karşılık gelen özellik değerleri arasındaki mutlak farkların toplamı olarak hesaplanır [23,24,25].

Çalışmada, elde edilen özelliklerin gruplanması için k ortalamalar yöntemi kullanılmıştır. Burada elde edilen küme merkezlerine yakınlığı ölçümlemek için Öklid uzaklık metriği tercih edilmiştir. Farklı veri setleri ve veri yapıları için farklı yöntemler de tercih edilebilir.

2.3. Veri kümesi ve içeriği

TVSum (TV Özeti), video özetleme araştırması için halka açık bir veri kümesidir. Video özetleme algoritmalarını değerlendirmek için akademik makalelerde yaygın olarak kullanılmaktadır. [26].

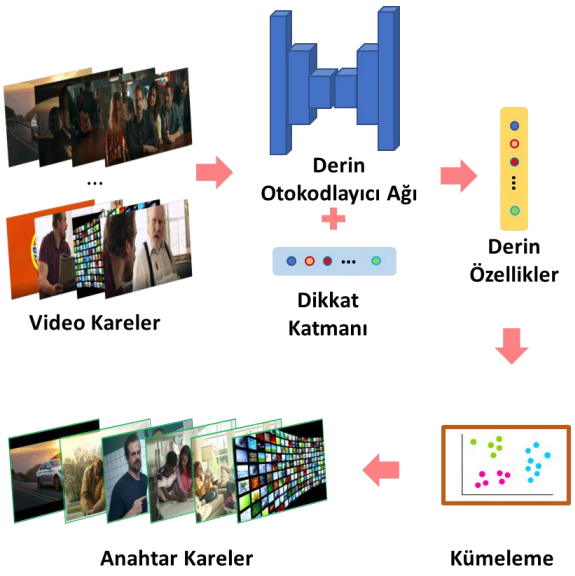
TVSum veri seti, haberler, talk şovlar, spor, belgeseller ve eğlence şovları dahil olmak üzere çeşitli türleri kapsayan 50 popüler TV şovundan video klipler içerir. Veri kümesi, video özetleme algoritmalarını farklı bağlamlarda değerlendirmek için uygun hale getiren çeşitli görüntüler içerir. Her video klip, değerlendirme için temel gerçek işlevi gören, insanlar tarafından oluşturulan özetlerle ilişkilendirilir. Veri kümesi aşağıdaki bileşenleri içerir:

Videolar: Veri kümesi, birkaç dakikadan birkaç saate kadar değişen uzunluklarda MP4 formatında video klipler içerir. Videolar, çok çeşitli konuları ve türleri kapsar ve video özetleme araştırması için çeşitli içerikler sağlar.

İnsan tarafından oluşturulan özetler: Her video, algoritmalar tarafından oluşturulan video özetlerinin kalitesini değerlendirmek için temel gerçek olarak kullanılan, insanlar tarafından oluşturulan birden çok özet ile ilişkilendirilir. Bu özetler, videoların ana içeriğinin kısa ve temsili açıklamalarını sağlar.

Video meta verileri: TVSum ayrıca her video için şov adı, bölüm başlığı, yayın tarihi ve video süresi gibi bilgiler dahil olmak üzere meta veriler sağlar. Bu meta veriler, bağlamsal analiz için veya videoları belirli kriterlere göre filtrelemek için kullanılabilir.

2.4. Test Ortamı Ve Önerilen Yöntem



Şekil 1:Uygulama akışı şeması

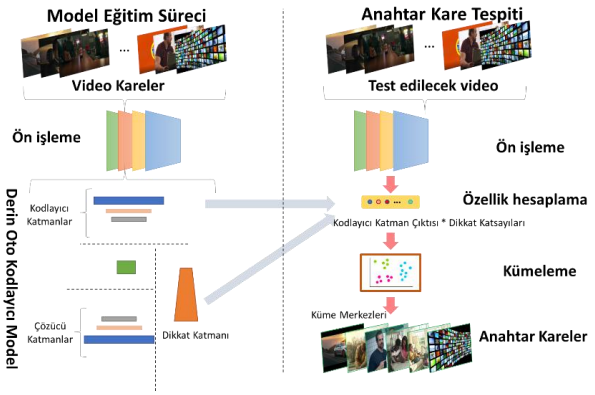
2.4.1. Model

Çalışmada bir oto kodlayıcı kullanılmıştır, girdi verilerini tipik olarak daha düşük boyutlu bir gizli alana sıkıştırarak ve ardından orijinal biçimine geri döndürerek yeniden yapılandırmak üzere eğitilmiş bir tür sinir ağıdır.

Model mimarisi bir kodlayıcı ve bir kod çözücüden oluşur. Kodlayıcı, (3, 64, 64) şeklindeki giriş verilerini alır, düzleştirir ve ardından 'relu' aktivasyon fonksiyonlarıyla iki(128,64) yoğun katmandan geçirilir. Sonrasında Kodlayıcının çıktısı, gizli katmanda 64 boyutuna sahip bir vektördür. Kod çözücü daha sonra sıkıştırılmış gösterimi alır ve bir relu aktivasyon fonksiyonuna sahip yoğun katmanlar(64,128) kullanarak orijinal giriş şekline geri döndürür.

Ayrıca modelin kodlayıcı kısmına dikkat katmanı eklenmiştir. Dikkat katmanı, kodlayıcının çıktısını alır, her zaman adımı için dikkat puanlarını hesaplar ve ardından bu puanları, dikkat ağırlıklı bir temsil oluşturmak için kodlayıcı çıktısına uygular. Ortaya çıkan dikkat ağırlıklı temsil daha sonra sabit boyutlu bir çıktı üretmek için küresel ortalama havuzlama katmanına (global average pooling) beslenir.

Çıktılar sonra "Adam iyileştirici"(adam optimizer) katmanına yönlendirilir ve optimize edilir. Son aşamada ortalama kare hata(MSE) kaybı işlevi ile hata hesaplaması yapılır ve güncellenir.



Şekil 2: Model şeması

2.4.2. Tespit katmanı

Bu adım, elde edilen model ve çıktıları değerlendirilmesi aşamasıdır. Eğitilen model veri özelliği çıkarmak için kullanılacaktır. Çıkarılan özellikler kümelenerek özellik grupları belirlenecektir. Video içeriğinden bu özeliklere en yakın kareler tespit edilerek anahtar kareler olarak işaretlenecektir.

İlk olarak, kareler HSV renk uzayına dönüştürerek, 64x64 piksel olarak yeniden boyutlandırarak ve piksel değerlerini 0 ila 1 aralığında normalleştirerek ön işleme tabi tutulur.

Ardından, özellikleri kümeler halinde gruplandırmak için K-Ortalamalar kümeleme algoritmasını kullanır. Bu uygulamada küme sayısı farklı sayılara ayarlanabilir, ancak uygulamaya bağlı olarak önceden belirlenmesi gerekir.

Kümeleme tamamlandıktan sonra algoritma, özellik mesafesi açısından küme merkezine en yakın kareyi seçerek her küme için anahtar kareyi tespit eder. Bu anahtar kareler, orijinal dizideki indeksleri ve özellik gösterimleriyle birlikte saklanır.

Genel olarak, bu yöntem, diğer karelere benzerliklerine dayalı olarak anahtar kareleri ayıklamanın basit ama etkili bir yoludur. Ortaya çıkan anahtar kareler, videoyu özetlemek veya içeriğinin temsili bir görsel özetini sağlamak gibi çeşitli amaçlar için kullanılabilir.

Burada kullanılan kümeleme algoritmalarının küme sayısı parametresi aynı zamanda çıkarılacak olan anahtar kare sayısını da belirlemektedir. Algoritmanın sayı parametresine bağımlılığı olduğundan her videoda aynı miktarda kare olduğu var sayılacaktır. Bu durumda daha durağan videolarda anahtar kare sayısı daha az olacağından benzer karelerin tespiti ile karşılaşılacaktır. Veya daha yüksek sayıda tespit edilmesi gereken durumlarda daha az sonuç üretilmiş olacaktır. Bu durumların etkisini daha azaltmak için tespit edilen karelerin sayısı genelde kullanılandan daha yüksek belirlenip benzer karelerin temizlenmesi işlemi yapılması uygun görülmüştür. Bu işlem için bir uzaklık metriği ve uzaklıklar matrisi kullanılacaktır. Tespit edilen uzaklıklara göre birbirine yakın olan karelerden biri bulgu listesinden silinecek ve diğer temsili tutulacaktır. Yakınlık sınırını belirlerken istatistiksel güven aralıkları göz önünde bulundurulacak ve ortalama değerinden iki standart sapma daha düşük uzaklıkta olan kareler birleştirilecektir.

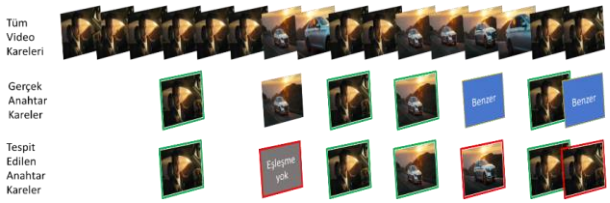
μ= $ortalama$

σ= $std.sapma$

$$\text{Uzaklık} < \mu - 2 * \sigma \text{ ise birleştir} \quad (1)$$

2.4.3. Ölçümleme metodları

Ölçümleme için kullanılan veri setinin anahtar kare düzeyinde bir etiket bilgisi olmadığından çıktının değerlendirmesinde uzman görüş kullanılacaktır. Anahtar kare olarak belirlenecek görüntünün video içerisinde yer alan her sahneyi tespit etmesi, eğer farklı sahneler yoksa görüntüdeki büyük değişiklikleri ve hareketleri tespit etmesi beklenecektir. Bu anlamda her video için anahtar kareler için kesit aralıkları belirlenecek ve test çıktısının bu aralıkta en az bir görüntü tespit edebilmiş olması beklenecektir. Elde edilecek veri ise her doğru tespit edilen ve tespit edilemeyen kareler için hata matrisi ile değerlendirilecektir.



*Şekil 3:*Tespit doğrulama durumları

Hata matrisi, bir makine öğrenimi modelinin performansını değerlendirmek için kullanılan bir tablodur. Bir ikili sınıflandırma problemi için gerçek pozitif (TP), gerçek negatif (TN), yanlış pozitif (FP) ve yanlış negatif (FN) değerlerinin bir matrisidir.

Matris tipik olarak dikey eksende gerçek sınıf etiketleri ve yatay eksende tahmin edilen sınıf etiketleri ile bir kare olarak temsil edilir.

Matristeki değerler, doğruluk, kesinlik, geri çağırma ve F1 puanı gibi bir makine öğrenimi modelinin performansını değerlendirmek için yaygın olarak kullanılan çeşitli ölçütleri hesaplamak için kullanılabilir. Bu metrikler, modelin ne kadar iyi performans gösterdiğine dair iç görüler sağlayabilir ve iyileştirme alanlarının belirlenmesine yardımcı olabilir.

3. Test

Tablo 1: Sınıf sonuçları hata matrisi

		gerçek	
		0	**1**
tahmin	**0**	**330**	**14**
	1	**12**	**43**

Tablo 2: Test sonuçları başarım skorları

Sınıf	kesinlik	duyarlılık	f1-skor
0	**0,96**	**0,96**	**0,96**
1	**0,78**	**0,75**	**0,77**

F1 skoru, sınıflandırma modellerinin performansını ölçmek için kullanılan bir ölçüttür. Bu skor, kesinlik (doğru pozitiflerin toplam pozitif tahminlere oranı) ve duyarlılık (gerçek pozitiflerin toplam gerçek pozitife oranı) metriklerini dengelemek amacıyla kullanılır. Modellerin hem yanlış pozitif hem de yanlış negatif sonuçları göz önünde bulundurarak performansını değerlendirir. Hangi metriğin daha etkin olduğu, uygulamanın gerekliliklerine bağlıdır. Bu anlamda çalışmada dengesiz sınıf problemi olduğundan, başarımını öncelikli olarak ölçmede F1 skoru kullanılması daha uygun olacaktır.

Model test edilirken çapraz doğrulama yöntemleri kullanılarak elde edilen başarımların ortalamaları değerlendirmeye dahil edilmiştir. Farklı model parametreleri ile deneme yapılsa da en başarımlı model yapısı sonuçlarda paylaşılmıştır.

Kesinlik, modelin pozitif örnekleri doğru bir şekilde sınıflandırma yeteneğini gösterir. Bu durumda, sınıf 0 için kesinlik 0,96'dır, yani model pozitif sınıflandırılan tüm örneklerin %96'sını doğru bir şekilde sınıflandırmıştır. Ancak anahtar kare tespiti yöntemleri doğası gereği dengesiz sınıf problemleri olduğundan, normal sınıf için değerlendirme yapmak anlamsız olacaktır. Bu sebeple tespit edilen karelerin doğruluklarını incelemek gerekir. Sınıf 1 için kesinlik 0,78'dir, yani model pozitif olarak sınıflandırılan örneklerin %78'ini doğru bir şekilde sınıflandırılmıştır. Duyarlılık, modelin pozitif örnekleri doğru bir şekilde tanımlama yeteneğini gösterir. Sınıf 1 için duyarlılık 0,75'tir, yani model sınıf 1'e ait tüm örneklerin %75'ini doğru bir şekilde tanımlanmıştır. F1 puanı, kesinlik ve duyarlılık arasındaki dengeyi gösteren bir ölçüttür. Bu durumda, sınıf 1 için F1 puanı 0,77'dir.

4. Sonuç

Bu çalışmada, dikkat katmanına sahip derin bir otomatik kodlayıcı kullanan bir anahtar kare çıkarma yöntemi önerilmiştir. Yöntem, önerilen yaklaşımın etkinliğini gösteren sınıflandırma görevinde 0,77'lik bir başarı oranı elde etmiştir.

Yöntem, önce kodlayıcı kısmını kullanarak video karelerinden özelliklerini çıkarır ve ardından K-means kümelemeyi kullanarak bu özellikleri kümeler. Anahtar kareler daha sonra, küme merkezine olan yakınlıklarına göre her bir kümeden seçilir. Otomatik kodlayıcının kodlayıcı kısmında dikkat katmanının kullanılması, video karelerindeki en belirgin özelliklerin vurgulanmasına yardımcı olur ve bu, anahtar kare çıkarma işleminin doğruluğunu artırmıştır.

Deneysel sonuçlar, önerilen yöntemin, anahtar kare çıkarma için mevcut yöntemler kadar iyi performans gösterdiğini ve video özetleme ve eylem tanıma gibi çeşitli alanlarda potansiyel uygulamalara sahip olduğunu göstermektedir. Sınıflandırma görevindeki 0.77 başarı oranı, önerilen yöntemin doğru ve etkili olduğunu göstermektedir. Farklı modellerin simülasyon sonuçlarının araştırılması, bu makalede önerilen modelin iyi bir performans gösterebileceğini ortaya koymuştur.

Ayrıca bazı önerilen yükse başarımlı yöntemler birden fazla modelin bir araya gelmesi şeklinde veya birden fazla ön işleme adımı ile tespit edilmiş özellik verileri ile çalışmaktadır. Bu durum günlük kullanım için yüksek işleme ve zaman maliyetleri yaratmaktadır. Önerilen model bu anlamda daha az kaynak ve zamana ihtiyaç duymaktadır.

Önerilen yöntem, genellikle buluşsal yöntemlere veya videonun anlamsal içeriğini yakalayamayan düşük düzeyli özelliklere dayanan mevcut anahtar kare çıkarma yöntemlerinin sınırlamalarının dışına çıkmaktadır. Dikkat katmanına sahip derin bir otomatik kodlayıcı kullanarak, yöntem video karelerindeki en göze çarpan bilgileri yakalayan üst düzey özellikleri ayıklayabilmektedir. Ayrıca denetimsizdir, bu da onu ölçeklenebilir ve çok çeşitli video analizi görevlerine uygulanabilir kılmaktadır.

Gelecekteki çalışmalarda, ses ve hareket bilgileri gibi ek özellikleri dahil ederek yöntemin performansını daha da geliştirmenin yolları keşfedilmesi planlanmaktadır. Yöntemin etkinliğini spor videoları, haber videoları ve gözetleme videoları gibi farklı video türleri üzerinde de araştırması planlanmaktadır. Önerilen yöntemin video analizinde anahtar kare çıkarma için umut verici bir çözüm sağladığı ve video özetleme ve video alma(retrieval) gibi çeşitli uygulamalara uygulanabileceği görülmektedir.

Sonuç olarak, dikkat katmanına sahip derin bir otomatik kodlayıcı kullanan önerilen anahtar kare çıkarma yöntemi, video analizi için umut verici bir yaklaşımdır ve çok çeşitli alanlarda potansiyel uygulamalara sahiptir. Bu yöntemin aktif bir araştırma alanı olmaya devam etmesi ve performansını ve doğruluğunu artırmak için daha fazla iyileştirme yapılabileceği düşünülmektedir.

Kaynakça

[1] Antani S., Xue L., and Thoma G.. Automatic key-frame extraction from video using hidden markov models. In Proceedings of the 20th IEEE International Conference on Image Processing (ICIP), pages 1081-1084, 2013.

[2] Xuelong Li, Bin Zhao, Xiaoqiang Lu, Xuelong Li, Bin Zhao, (2017) “Key Frame Extraction in the Summary Space” PMID: 28693004

[3] Chen K., Huang K., and Chen T.. An unsupervised approach to video keyframe extraction based on objectness measure. IEEE Transactions on Multimedia, 20(5):1121-1135, 2018.

[4] Liu Y., He L., Luo M., and Wu Y.. Deep learning for video summarization: A review. ACM Transactions on Multimedia Computing, Communications, and Applications, 16(3s):1-22, 2020.

[5] Lu H., Zhang C., and Li H. Saliency detection based on attention mechanism: A survey. IEEE Transactions on Circuits and Systems for Video Technology, 30(11):4217-4240, 2020.

[6] Wang J., Song Y., Leung T., Rosenberg C., Wang J., Philbin J., Chen B., and Wu Y.. Learning fine-grained image similarity with deep ranking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1386-1393, 2014.

[7] Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. Science, 313(5786), 504-507.

[8] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press.

[9] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.).

[10] Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008). Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning (pp. 1096-1103).

[11] Zhang, H., Goodfellow, I., Metaxas, D., & Odena, A. (2019). Self-Attention Generative Adversarial Networks. arXiv preprint arXiv:1805.08318.

[12] Woo, S., Park, J., Lee, J. Y., & Kweon, I. S. (2018). CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 3-19).

[13] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).

[14] Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., & Sun, J. (2019). DetNAS: Neural Architecture Search on Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 11281-11289).

[15] Zhang, S., & Patel, V. M. (2020). Deep learning for neuroscience. Nature Neuroscience, 23(7), 811-821.

[16] Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. ACM Computing Surveys, 31(3), 264-323.

[17] Ng, A. Y., Jordan, M. I., & Weiss, Y. (2002). On spectral clustering: Analysis and an algorithm. In Advances in neural information processing" systems (pp. 849-856).

[18] Arthur, D., & Vassilvitskii, S. (2007). k-means++: The advantages of careful seeding. In Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms (pp. 1027-1035). Society for Industrial and Applied Mathematics.

[19] Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (pp. 226-231).

[20] Jain, A., & Dubes, R. (1988). Algorithms for clustering data. Prentice-Hall.

[21] Aggarwal, C. C., & Reddy, C. K. (2013). Data clustering: algorithms and applications. Chapman and Hall/CRC.

[22] Singh, A. K., & Yadav, A. (2017). A comparative study of distance measures in clustering. International Journal of Advanced Research in Computer Science, 8(3), 403-407.

[23] Endres, D. M., & Schindelin, J. E. (2003). A new metric for probability distributions. IEEE Transactions on Information Theory, 49(7), 1858-1860.

[24] Yang, L., Jin, R., & Sukthankar, R. (2006). Detecting faces in images: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(1), 34-58.

[25] Manning, C. D., Raghavan, P., & Schütze, H. (2008). Introduction to information retrieval. Cambridge University Press.

[26] Mahasseni, B., Lam, M., Tavakoli, H. R., & Fernando, B. (2017). Unsupervised video summarization with adversarial LSTM networks. In Proceedings of the IEEE International Conference on Computer Vision (pp. 4085-4094).