

Kaba Faksimile Verilerinin Gönderilmesinde Gereksizlik Azaltıcı Basit Algoritmalar*

Yazan :
Peter A. STERN
SIEMENS AG, MÜNİH

Çeviren:
Dr. Güney GÖNENÇ
ODTÜ

ÖZET

Daktilo yazısı harfler için tarama sıklığını azaltan fakat ayrıntıların kaybolmasına yol açmayan basit algoritmalar verilmiştir. Tarama sıklığı yeteri kadar yüksekse faksimile iletim hızının nasıl 4 kat azaltılabildiği bilgisayar benzetimi ile gösterilmiştir.

SUMMARY

Simple algorithms are introduced with which one can reduce the resolution of type-written letters without eliminating isolated details. Computer simulations show how the transmission rate of facsimile can be reduced by a factor 4 given a sufficiently high resolution.

1. GİRİŞ

Daktilo yazısının pratik sayısal (digital) faksimilesi (tıpkıbasım) en azından mm başına 5 çizgilik bir tarama sıklığı gerektirir. Böylece 0,2 mm genişliğindeki ince ayrıntıların gönderilmesi ve alıcıda belirlenmesi mümkün olur. Ne var ki böyle ince ayrıntılar ancak arasıra ortaya çıkmaktadır. Böylece, tarama sıklığının yatay ve düşey doğrultularda yarıya indirilmesi ve böylece bit hızının 1/4 e indirilmesi olanaklı görünmektedir. Ancak genellikle böyle «zorlamayla» yapılan azaltmalarda bazı kapalı çevrelerin tanınması için gerekli olan ince ayrıntılar kaybolacaktır. Örneğin konumu, özgün (original) kafesteki atlanan çizgilerden birinin üstüne rastlamış 0,2 mm kalınlıkta bir ince çizgi böyle-

ce kaybolur. Karakter tanıma ile ilgili olarak bu konu Endres'ce tartışılmıştır [1].

Bu yazıda tartışılacak olan yöntemler, her iki doğrultuda 2 oranında tarama azaltılmasına (ve sonuç olarak 4 oranında bit hızı azaltılmasına) olanak veren, ama arasıra ortaya çıkan önemli ince ayrıntıların yok olmasına da izin vermeyen algoritmalar olacaktır. Bu ayrıntılar bir miktar kalınlaştırılmış ve kaydırılmış olacaklardır, ne var ki bu, çoğu kez sakıncalı olmayacaktır. İnce bir ayrıntının kalınlaştırılmasına, elbette ancak bu ayrıntı civardaki başka ince ayrıntılarla karışmıyor ise izin verilebilir. Örneğin, 0,2x0,2 mm lik bir satranç tahtası örüntüsü bu koşullar altında ayırdedilemez duruma düşer, ancak bu türden ince ayrıntılar daktilo yazısının gönderilmesinde gereksizdir. Ayrık durumdaki ince ayrıntıların kalınlaştırılıp kaydırılması, gereksiz-

(*) Nachrichtentechnische Zeitschrift, Cilt 26, Sayı 9 (Eylül 1973), s. 417-420.

lik azaltılmasının görünür bir biçimde ortaya çıkması olmaktadır.

Daktilo yazısı harflerdeki (küçük harflerin ortalama yüksekliği yaklaşık 2,5 mm) gereksizliğinin, harflerin okunabilirliğini ortadan kaldıracak kadar çok bozulma (distorsiyon) yaratmaksızın, azaltılabileceği bilgisayar benzetimi (computer simulation) ile görülmüştür. Öte yandan «zorlama» yöntemiyle yapılan bir azaltma, bazı harfleri tanınmayacak hale getirebilen bozulmalara yol açabilmektedir.

Bu yazıda anlatılan yöntemlere bazı bakımlardan benzeyen bir deterministik gereksizlik azaltma yöntemi P. Stucki [2] tarafından verilmiştir. Stucki'nin yöntemi ana olarak şudur: Her satırda noktalar birer atlayarak örnekleme yapılır. Atlanmış noktalar, alıcı tarafta, alman noktalar üstüne özel bir «kalıp» (şablon) yerleştirilmek ve atlanmış noktayı eğer örüntüye (pattern) uyuyorsa varsaymak yoluyla doldurulur (Stucki bu kalıplara «hata trominolan» adını veriyor). Böylece elde edilen azalma 2 oranındadır. Alıcıda bir gecikme devresi gerekli olmaktadır. Keskinlik (resolution) azalması doğuran başka gereksizlik azaltma yöntemleri [3] ve [4] te verilmiştir.

2. ALGORİTMALAR

Özgün tarama kafesi ve alıcıdaki tarama kafesi sırasıyla Şekil 1a ve 1b de görülmektedir. Algoritmaları açıklayabilmek için özgün kafeste 7 noktayı P_i ile gösterelim ($i = 0,1, \dots, 6$). P_0 tarayıcı kafanın şimdiki (içinde bulunulan andaki) konumu olsun. Demek ki P_i aynı tarama satırındaki bir önceki nokta ve P_3 bir önceki satırdaki komşu noktadır. Alıcı tarama kafesinde P_i lere karşılık gelen noktalar Q_i noktalarıdır ($i = 0,1, \dots, 6$). P_i ve Q_i ikili fonksiyonlardır: Örneklenen nokta karaysa bu fonksiyonun değeri 1, beyazsa 0 dır.

Algoritmalara geçmezden önce iki önemli kabul yapacağız.

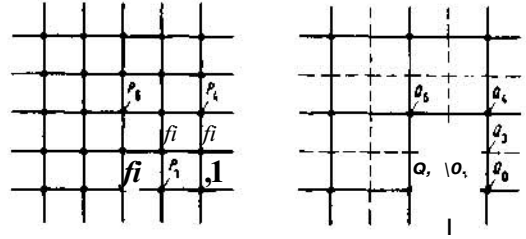
1. Satırlar birer atlanarak, her satırda da noktalar birer atlanarak gönderilecektir.

2. Kanalda hata söz konusu değildir.

Demek ki Şekil 1b deki alıcı kafesinde çift sayılı Q_i ler gerçekten gönderilen noktalara karşılık gelen noktalardır. Bu noktalar her iki doğrultuda da birer satır atlanarak elde edilen ve şekilde kalın siyah çizgilerle gösterilen «indirgenmiş kafesin üzerinde bulunurlar. Kanalda hata olmadığından alınan ve gönderilen noktalar (çift sayılı Q_i ler) birbirinin aynısıdır. Tek sayılı Q_i ler gönderilmeyen (demek ki alınmayan) ara noktalardır ki bunların, tarama kes-

kinliğinin özgündekinin aynısı olması için, doldurulması gerekmektedir.

→ Tarama doğrultusu b)



Şekil 1. Tarama kafesleri, (a) özgün, (b) alıcı.

Alıcıda, yazıcı kafa, aslında bütün kafesteki her noktaya dokunmaktadır; ama kafa sadece indirgenmiş kafesteki noktaların (çift sayılı Q_i ler) üstünden geçerken bir bitlik bilgi (information) alınmaktadır. Tek sayılı Q_i ler için bir renk saptanması gereklidir. Tarayıcı kafa (özgün kafeste) P_0 da iken yazıcı kafa Q_0 da dahil önceki bütün noktaları doldurabilir. Kara mı beyaz mı olacağını şimdilik belirtmeksizin iletilmeyen Q_1, Q_3, Q_5 noktalarının Q_0 la aynı renkte olduklarını kabul edeceğiz:

$$Q_1 = Q_3 = Q_5 = Q_0 \quad (D)$$

Bu dört nokta için «indirgenmiş kare»nin köşelerini oluşturuyor diyeceğiz.

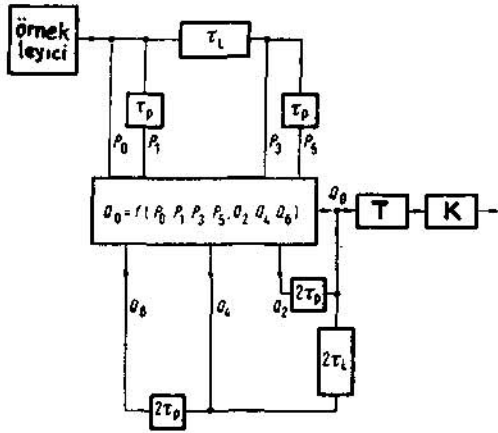
Geriye, Q_0 in ve sonuç olarak indirgenmiş karenin öteki üç köşesinin hangi renkte olacağına karar vermek kalıyor. Bu karar veren algoritmalar iki türe ayrılabilir. Birinci türden algoritmalarda, Q_0 sadece indirgenmiş karedeki dört örnekleme noktasının bir fonksiyonudur:

$$Q_0 = f(P_0, P_1, P_3, P_5) \quad (2)$$

İkinci türden algoritmalarda, Q_0 sadece bu dört noktanın değil, fakat çevredeki iletilen noktaların da (indirgenmiş kafeste) fonksiyonudur:

$$Q_0 = f(P_0, P_1, P_3, P_5; Q_2, Q_4, Q_6) \quad (3)$$

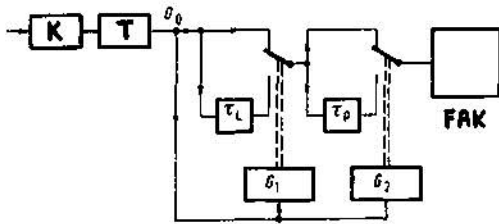
Şekil 2 de bu eşitliğe uygun verici görülmektedir. P_i ler özgün örnekleme noktalarıdır; Q_i ler ise iletilen noktalar olup bunlar hem de $f(\cdot)$ fonksiyon üreticisine geri beslenmektedirler. T_D özgün kafese ilişkin nokta gecikmesi, t_L ise özgün kafese ilişkin satır gecikmesidir. Örneğin P_5, P_0 a göre bir satır periyodu ve bir nokta periyodu kadar gecikmelidir (Şekil 1a). Tampon (buffer) devresi çıkış bit hızını giriş bit hızının 1/4 üne ayarlar.



Şekil 2. (3) eşitliğine göre verici.

T_p : nokta gecikmesi, T_L : satır gecikmesi,
 T : tampon (buffer), K : kanal.

iletilmeyen noktaları da (tek sayılı Q_i ler) dolduran alıcı Şekil 3 te görülmektedir. T_D ve T_L vericideki periyodların aynısıdır. G_j üretici ilgili anahtar $1/2T_L$ frekansında, G_2 üretici de $1/2x_p$ frekansında kapatıp açar. Özgün kafesteki P_0 (Şekil 1a) ile alıcı kafesteki O_0 (Şekil 1b) bir birine karşılıklı gelmekte iseler de, yazıcı kafa tarayıcı kafaya göre bir satır artı bir periyod gecikmelidir. Böylece Q_0 iletilen noktası alındığında (Şekil 3), bir yandan hemen FAK çıkışına aktarılmakta ve Q_5 noktası olarak yazılmakta (Şekil 1b), öte yandan bir örnekleme periyodu kadar geciktirilip Q_3 olarak yazılmaktadır. Bu nokta tüm bir satır kadar daha geciktirilip Q_1 olarak, ilâveten bir örnekleme periyodu kadar daha geciktirilip Q_0 olarak yazılır. «İndirgenmiş kare»nin dört köşesi böylece (1) eşitliğine göre doldurulmuş olur.



Şekil 3. Alıcı.

G_1 : satır frekansının yarı değerini üretir (alınan her satır bir kez tekrarlanır),
 G_2 : nokta frekansının yarı değerini üretir (alınan her nokta bir kez tekrarlanır),
 T : tampon (buffer), K : kanal, FAK : faksimile çıkışı.

Birinci türden algoritmalara (2 eşitliği) örnek olarak şu özel durumu gözönüne alıyoruz :

$$Q_n = P_0 \quad (4)$$

Bu algoritma tarama kafesinin her iki doğrultuda yarıya indirilmesi olduğundan (örnekleyici, noktaları ve satırları birer atlayarak almaktadır), buna «zorlama» azaltma yöntemi diyoruz. İlerde göreceğimiz gibi bu algoritma ince ayrıntıların kaybolmasına yol açmaktadır. Örneğin, eğer beyaz zemin üzerinde P_0 ile P_2 arasındaki P_j den geçen çok ince bir kara çizgi varsa, o zaman (4) algoritmasında bu çizgi yok olacaktır. Bu durumda, P_1 noktası, P_0 ve P_2 ye göre bir «tekil nokta» olmuş olur. Böyle «tekil nokta»ların varlığı halinde ince ayrıntıların kaybolmasını engellemek üzere, ikinci türden algoritmaların bir özel halini ortaya atacacağız.

Alıcıda ortaya konan şeklin sadece özgün noktaların değil, iletilen noktaların da bir fonksiyonu olması halinde (3 eşitliği), «tekil nokta» yi şöyle tanımlayacağız : Eğer bir önceki iletilen Q_2 noktası ile, özgün kafesteki şu anda örneklenen P_0 noktası aynı renkte iseler ve bir önceki örneklenen P_1 noktasının rengi P_0 m renginin tersi ise yatay doğrultuda bir «tekil nokta» vardır diyeceğiz. Matematiksel olarak,

$$P_0 = Q_2 \text{ ve } P_1 = \bar{P}_0 \quad (5a)$$

ise «yatak tekillik» vardır diyoruz.

Benzer şekilde ,eğer,

$$P_0 = Q_4 \text{ ve } P_3 = \bar{P}_0 \quad (6a)$$

ise «düşey tekillik» ve,

$$P_0 = Q_0 \text{ ve } P_5 = \bar{P}_0 \quad (7a)$$

ise «köşegenel tekillik» vardır diyeceğiz.

Şimdi, ikinci türden algoritmaların (3 eşitliği) bir özel hali olarak aşağıdaki algoritmayı ortaya atıyoruz: Eğer bir yatay tekillik varsa, şimdiki iletilen nokta Q_0 , tekil nokta P_j ile aynı renkte (değerde) olacak; eğer yatay tekillik yoksa, Q_0 , şimdiki örneklenen nokta P_0 ile aynı renkte olacak. Matematiksel olarak, eğer (5a) doğruysa, o zaman :

$$Q_0 = P_j \quad (5b)$$

eğer (5a) doğru değilse, o zaman :

$$Q_0 = P_0 \quad (5c)$$

olacak.

Benzer şekilde, eğer bir düşey tekillik varsa, yani (6a) eşitliği doğruysa, o zaman :

$$Q_0 = P_j \quad (6b)$$

t,ba) eşitliği yanlışsa, o zaman :

$$0_{\cdot} = P_{\cdot} \quad (6c)$$

olsun Eğer köşegenel tekillik varsa, yani (7a) eşitliği doğruysa ,o zaman :

$$0_{\circ} = P_{\circ} \quad (7b)$$

(7a) eşitliği yanlışsa, o zaman :

$$0_{\circ} = P_{\circ} \quad (7c)$$

olsun.

Yukardaki üç tipten tekilliğin iki çeşit birlikte olma durumunu içeren özel ikinci tür algoritmaların benzetimi yapılmıştır. Bu algoritma şöyledir :

a. Yatay ve düşey tekilliklerin herhangi birinin ya da ikisinin birden bulunması durumu. Eğer (5a) ya da (6a) eşitliklerinden biri ya da her ikisi birden doğru ise (mantıksal «ya da» işlemi) o zaman iletilen nokta Q_{\circ} , P, noktasına ya da P, noktasına ya da her ikisine birden eşit olsun. Mantıksal bir bağıntı olarak bunu şöyle belirleyebiliriz:

Eğer (5a ya da 6a) doğru ise, o zaman:

$$\left. \begin{array}{l} Q_{\circ} = (P, \text{ ya da } P_{\circ}) \\ \text{aksi halde :} \\ 0_{\circ} = P_{\circ} \end{array} \right\} (8)$$

b. Yatay, düşey ya da köşegenel bir tekilliğin, ya da bunlardan herhangi ikisinin, ya da üçünün birden bulunması durumu.

Eğer (5a ya da 6a ya da 7a)doğru ise, o zaman:

$$\left. \begin{array}{l} Q_{\circ} = (P, \text{ ya da } P_3, \text{ ya da } P_5) \\ \text{aksi halde :} \\ Q_{\circ} = P_{\circ} \end{array} \right\} (9)$$

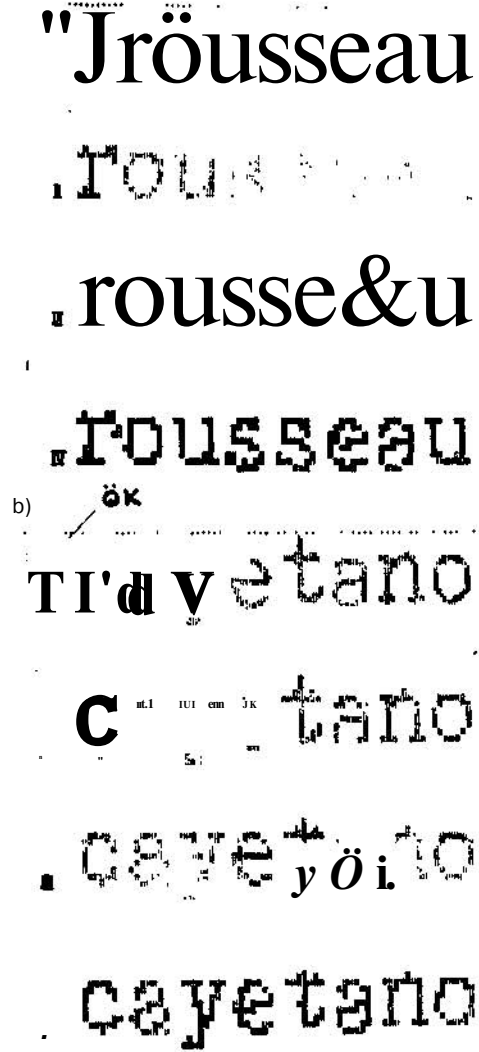
Eğer P_j , P_3 ve P_5 sayfa üzerinde zaman zaman bulunabilecek olan gerçekten tekil noktalar ise, o zaman bunlar algoritma etkisiyle büyütülecek ve biraz kaydırılacaklar, fakat kaybolmayacaklardır, öte yandan, bir doğrultudaki noktalar (diyelim ki yatay), ince bir satranç tahtası örüntüsü biçiminde, bir siyah bir beyaz sıralanmış iseler, o zaman alman şekil aynı doğrultuda üç katı büyümüş bir satrançlı örüntü olacaktır.

3. BİLGİSAYARLA BENZETİM VE SONUÇLARI

Birinci türden algoritmalarla örnek olarak (4) eşitliğini, ikinci türden algoritmalarla örnek olarak da (8) ve (9) eşitliklerini benzetimledik. Şekil 4 (1) de «cayetano» sözcüğünü oluşturan

daktilo küçük harflerinin, bilgisayarla benzetimlenmiş çıktıları görülmektedir. Örnekleme inceliği 6 2/3 satır/mm ye eşdeğerdir. Özgün örnekleme kafesinin bir parçası da şekilde gösterilmiştir

Aynı harflerin «zorlama» (4 eşitliği) yöntemiyle gereksizlik azaltılmasına uğratıldıktan sonraki durumları, Şekil 4 (II) de görülmektedir. Bilgi kaybı örneğin «a» harflerinde görülmektedir. Kapalı kısımlar açık hale gelmiştir, üstelik birinci «a» da açık bir kısım kapanmıştır. Aslında, bu harfin artık «a» mı yoksa «e» mi olduğu anlaşılamamaktadır.



Şekil 4. Çeşitli algoritmaların uygulandığı daktilo yazısı sözcükler (bilgisayar benzetimi ile elde edilmiştir). ÖK: örnekleme kafesi.

- I. Özgün sözcük (original),
- II. «zorlama» yöntem (4 eşitliği),
- III. (8) eşitliğine uygun yöntem,
- IV. (9) eşitliğine uygun yöntem.

Bilgi kaybı, veya ince ayrıntıların yokolması, yatay ya da düşey doğrultuda tekilliklerin varlığını gözönüne alan algoritma (8 eşitliği) ile azaltılabilir. Şekil 4 (III) de her iki «a» harfinin de alt tarafları gerektiği gibi kapanmış durumdadır, fakat üst tarafları halâ kapalıdır. Yatay ve düşey tekillikler yanında köşegenel tekillik de gözönüne alınır (9 eşitliği), harflerin çevrelerinde iyileşme olmaktadır: Artık kapalı olması gereken çevreler kapalı, açık olması gerekenler açıktır. Şekil 4 (IV) te bu durum görülmektedir. Burada, yalnızca harflerin köşelerinde bir düzensizlik vardır, bu da tüm yazıya biraz bulanık bir görünüm vermektedir. Tarama inceliği 6 2/3 satır/mm den daha büyük olmuş olsaydı köşelerdeki bu düzensizlik yokolacaktı, ama o zaman daha yalın olan algoritmanın (8 eşitliği), verdiği sonuçlar da iyileşmiş olacak ve daha karmaşık algoritmaya (9 eşitliği) gerek kalmayacaktı. Bu nedenle P_0 in çarpaz olarak üstünde ve sağındaki (Şekil 1a) tekillik gibi tekillikler gözönüne alınmamıştır.

4. SONUÇ

Gönderilen noktanın geri beslenmesine (3 eşitliği) dayanan iki algoritmanın (8 ve 9 eşitlikleri), örnekleyici sıklığının yeteri kadar olması ve ince ayrıntıların sadece ara sıra ortaya çıkması halinde, daktilo yazılarının gönderilmesinde gönderme hızının azaltılması yönünden etkili olduğu gösterilmiştir. Bu iki algoritmayla elde edilen sonuçlar tarama sıklığına bağlıdır; (9) eşitliğiyle daha fazla tarama sıklığı gerekmektedir, fakat tarama sıklığı arttılınca da (8) eşitliği yeterli duruma gelmektedir ve başka tekilliklerin hesaba katılmasına gerek kalmamaktadır. Gönderme hızında, böylece, 4 kat bir azalma sağlanmaktadır.

Bu sonucu Stucki'nin yöntemiyle karşılaştırmak ilgi çekicidir. Örüntünün yeniden yapımı, bu yöntemde, alıcıda gerçekleştirilmektedir. Oysa bu iş, burada anlatılan yöntemde vericide gerçekleştirilmektedir. Buradaki yöntem, meteoroloji hizmetleri gibi tek verici fakat çok sayıda alıcı kullanan faksimile (tıpkıbasım) sistemleri açısından, daha üstündür [5]. Stucki yönteminde, geribesleme sözkonusu olmadığından, ince ayrıntılar genellikle kaybolmakta ve 2 oranında

bir azalma elde edilebilmektedir; alıcıda bir gecikme devresi gereklidir. Bizim yöntemimizde 4 oranında bir azalma sağlanmaktadır ve ince ayrıntılar kaybolmamakta sadece bozulmaktadır. Ancak daha fazla donanım gereklidir: Vericide uç gecikme ve bir tampon (buffer) devresi, alıcıda bir gecikme ve bir ters tampon devresi gerekli olmaktadır. Stucki yönteminde ise tampon devresi gerekmemektedir.

Teşekkür :

Yazar, yardımlarından ötürü Dr. W. E. Heinlein'e teşekkürü borç bilir.

KAYNAKLAR

1. *Endres, W.:* Über die Redundanzverminderung bei der Zeichenerkennung. Nachrichtentechn. Z., Cilt 16 (1963), s. 529-535.
2. *Stucki, P.:* Efficient transmission of graphics using polyomino filtering at the receiver. IEEE Trans. Aerospace and Electronic Systems, AES-6, (1970 Kasım), Sayı 6, s. 811-814.
3. *Smith, I. R.; Schilling, Donald L.:* Compressions of Bandwidth requirements for a certain class of band limit functions. IEEE Trans. COM-20 (1972 Nisan) Sayı 2, s. 104-114.
4. *Spencer, D. R.; Huang, Thomas:* Bit plane encoding of continuous pictures. Symp. Computer Processing in Communications, Polytechnic Inst. of Brooklyn, 1969 Nisan, s. 101-120.
5. *Straiton, J. C.:* Weather Bureau facsimile: Today, the future, problems. Proc. IEEE Internat. Convention on Communications, Boulder Colo., 1969, s. 11.11-11.14.