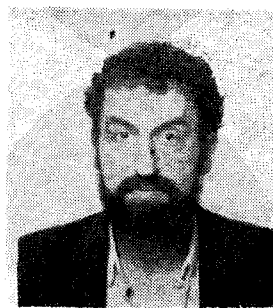


A VERBIDENT-SD-2 izolált szavas gépi beszéd felismerő

DR. FARAGÓ ANDRÁS—DR. GORDOS GÉZA—
—KOUTNY ILONA—MAGYAR GÁBOR—
DR. OSVÁTH LÁSZLÓ
BME Híradástechnikai Elektronika Intézet
TAKÁCS GYÖRGY
Posta Kísérleti Intézet



ÖSSZEFOGLALÁS

A dolgozat egy izolált szavas, személyfüggő, kötött szótáras beszéd felismerő használatát és működési algoritmusát írja le. A beszéd felismerő egy IBM—PC típusú személyi számítógépen, egy abba bedugható digitális jelfeldolgozó kártyán és megfelelő szoftveren alapul.

1. Bevezetés

Napjainkban a digitális beszéd feldolgozás eljutott arra a szintre, hogy a laboratóriumok világából kilépve a nagyközönség számára is használható és elérhető szolgáltatásokat nyújtson. Ennek egyik jelentős — és tudományos szempontból is izgalmas — területe a gépi beszéd felismerés, amely kulcsfontosságú az ember-gép kommunikáció beszéd útján történő megvalósításában. A gépi beszéd felismerés komoly segítséget nyújthat az olyan munkafolyamatokban, amelyekben a munkát végző embernek úgy kell bizonyos funkciókat ellátnia, hogy közben foglalt a keze (pl. orvos a műtőben, pilóta, minőségellenőr, az autó vezetése közben telefonáló stb.).

A gépi beszéd felismerő rendszerek viszonylag legegyszerűbb változata az, amelyik a felismerendő szavak izolált ejtését követeli meg és a szó csak egy — a felhasználó által előre bementett — korlátozott méretű szótár eleme lehet. Egy ilyen — izolált szavas, személyfüggő — beszéd felismerőt fejlesztett ki az OMF B támogatásával a Budapesti Műszaki Egyetem Híradástechnikai Elektronika Intézete (BME—HEI), együttműködve a Megamicro Kiszövetkezettel és az MTA Akusztikai Kutatólaboratóriumával.

A rendszer fantázia neve VERBIDENT—SD—2 (SD=Speaker Dependent, személyfüggő). Felépítésének alapvető jellemzője, hogy nem különállóan működő berendezésről van szó, hanem az IBM—PC (XT vagy AT) típusú személyi számítógép egy digitális jelfeldolgozó kártyával együttműködve végzi a beszéd felismerést. Megjegyzendő, hogy ez a kártya, amely TMS 32010 jelfeldolgozó processzort tartalmaz, nemcsak beszéd felismerésre alkalmas, hanem általános célú digitális jelfeldolgozásra is. Nagy sebessége teszi lehetővé a felismerési feladat gyakorlatilag real-time megoldását.

A rendszer szoftver szempontból két alapvető részre tagolódik. Az egyik a felismerő program, amely — a felhasználó által korábban már meg szerkesztett szótárak segítségével — a tényleges

DR. FARAGÓ ANDRÁS

1976-ban szerzett villamosmérnöki oklevelet a BME Villamosmérnöki Karának Híradástechnikai szakán. 1982-ig ugyanott a Matematika Tanszéken dolgozott tanársegédként, majd a BME Híradástechnikai Elektronika Intézetébe

került, ahol jelenleg adjunktusként dolgozik az Átvitel- és Rendszer technika Osztályon. A villamosmérnök hallgatók oktatása mellett fő szakmai érdeklődési és kutatási területe a digitális beszéd feldolgozás. Ezenkívül foglalkozik matematikai kutatással is, a gráfelmélet és kombinatorikus optimalizálás területén.

beszéd felismerést végzi. A másik alapvető szoftver komponens a szótárszerkesztő program, amellyel a felhasználó tetszése szerinti szótárat szerkeszthet, mágneslemezen tárolhat, módosíthat, továbbá speciális szolgáltatásokat (pl. spektrogram rajzolás) is igénybe vehet.

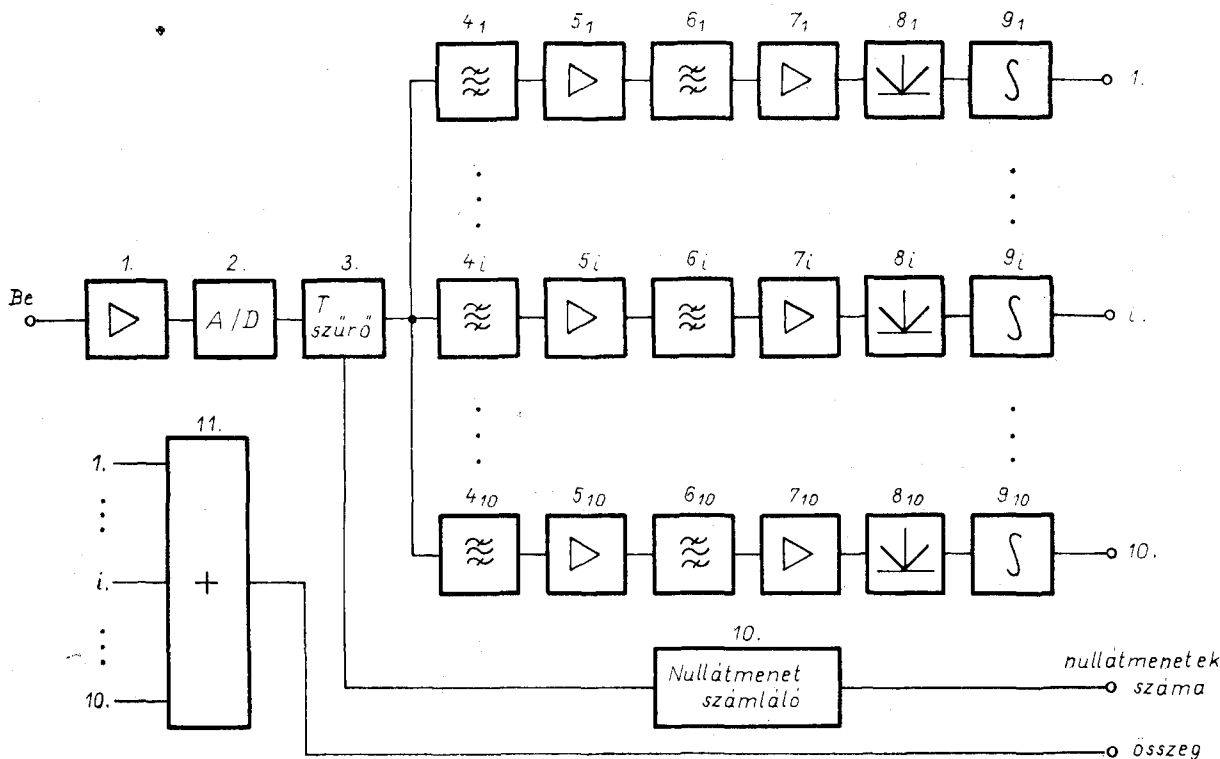
A rendszer hardware felépítését az alkotók külön dolgozatban tervezik megjelentetni. A jelen dolgozat a működési algoritmust írja le.

2. A rendszer használata és működésének főbb lépései

A rendszer bejelentkezéskor megkérdezi a felismeréshez használt referenciaszótár nevét, majd betölti azt a memóriába. A szótárt a felhasználónak előzőleg el kellett készítenie a saját hangjára, ha ilyen nincs, kijelzi. (Megjegyezzük, hogy tetszőleges számú felhasználó használhatja a rendszert, ha létrehozza a szótárat.) Ettől kezdve a felismerő automatikusan dolgozik.

A bemondás kezdete nincs időhöz vagy nyomógombhoz kötve, a rendszer folyamatosan bevételez. Ha szót érzel, fűttyszóval jelzi, s megkezd a feldolgozást, először az energia- és időnormálást. Ha a bemondott szó és a referenciaszó hossza között az eltérés kettes faktornál kisebb, akkor dinamikus programozással távolságukat kiszámolja. Az osztályozó algoritmus ezután kiválasztja a tesztszóhoz legközelebb álló referenciaszót. Ezt újabb fűttyszó követi, majd az eredmény megjelenik a képernyőn. Ha bizonyos feltételek nem teljesülnek, a rendszer a bemondás megismétlését kéri. A felismerés reakcióideje függ a szótár nagyságától; szótárelemenként mintegy 50...70 millisekundumot igényel. A kísérletezés fázisában több változat készült az értékelés nyomon követhetősége céljából. Például kérésre kiírhatók a különböző referencia elemektől mért távolságok. Kiraj-

Beérkezett: 1987. XI. 18. (..)



H-399-1

1. ábra. A lényegkiemelés elvi blokkvázlata. (A műveleteket a digitális jelfeldolgozó egység végzi, időosztásban)

zoltatható egymás mellé a bemondott szó és a felismerni vélt referenciaszó spektruma. Így képet kaphatunk az idő- és energia viszonyokról.

Egy másik változat nyugtázást tartalmaz. Így minden tesztszó után a JÓ, ISMÉTLÉS vagy VÉGE szavakat várja a rendszer. A jól felismert szavakról statisztikát készít. Itt a rendszer leállítása is beszéddel („VÉGE”) történik.

3. A felismerés működési algoritmus

Az alábbiakban összefoglaljuk a felismerési folyamat fő lépéseit. Ezek részletesebb általános kifejtése magyar nyelven is hozzáférhető (ld. GORDOS—TAKÁCS [1]). Itt most mindenekelőtt a VERBIDENT—SD—2 rendszer sajátosságait emeljük ki.

3.1. Lényegkiemelés

A lényegkiemelés alapvető feladata az, hogy a beszédjelet — mintavételezés és kvantálás után — leíró adattömeget valamiképpen redukálja, kevesebb — de a lényeg megőrző — adattal reprezentálja. A VERBIDENT—SD—2-ben ezt az ún. sávszűrős lényegkiemeléssel oldjuk meg, amely lényegében a beszédjel időben változó („gördülő”) spektrumának egy közelítését adja.

A sávszűrős rendszerű lényegkiemelés teljes egészében az IBM—PC-be bedugandó, TMS processzorral működő digitális jelfeldolgozó kártya

végzi. A jelfeldolgozás teljes folyamatát az 1. ábra mutatja, tekintet nélkül arra, hogy az adott funkció hardver vagy szoftver úton valósul meg.

A mikrofon által szolgáltatott jelet a jelfeldolgozó kártyán elhelyezett bemenő erősítő (1) felerősíti, úgy, hogy a jel szélső értékei hangos beszédben még éppen ne vezéreljék túl a (2) analóg-digitál átalakítót. A felerősített jeltől 10 kHz gyakorisággal vett mintákat az A/D átalakító 12 bites, kettes komplement kódú számokká alakítja. A maximális, +2,5 V-os feszültségnek így a 07FF hexadecimális szám felel meg, a TMS processzor 16 bites számábrázolásában.

A szoftver úton végrehajtott jelfeldolgozás első lépése egy

$$H_s(Z^{-1}) = 1 - 2Z^{-2} + Z^{-4}$$

átviteli függvényű transzverzális szűrés (3). Ez a tényező ugyanis valamennyi sávszűrő átviteli függvényében előfordul, és így valamennyi szűrőágra nézve azonos feldolgozást jelent.

A közös transzverzális szűrővel megszürt beszédjel mintáit a következőkben 10 db all-pole szűrővel dolgozza fel, így voltaképpen — a transzverzális szűrővel együtt — 10 db sávszűrő alakul ki. Itt jegyezzük meg, hogy valamennyi sávszűrő negyedfokú, a keresztvezési csillapításuk 3 dB, s az átviteli függvényük maximális laposságú.

A szűrők adatai az 1. táblázatban láthatók. Az i -edik all-pole szűrő ($4i-6i$) átviteli függvénye

A VERBIDENT—SD—2 lényegkiemelőjében szoftver úton megvalósított digitális sávszűrők egyes adatai. (Frekvencia-adatok kHz-ben)

| Sorszám | Sávhatárok (3dB) | Sávközép | | Sávszélesség | |
|---------|---------------------|----------|---------|--------------|---------|
| | | számtani | mértani | abszolút | relatív |
| 1. | 0,17—0,3 | 0,235 | 0,226 | 0,13 | 1,76 |
| 2. | 0,3 —0,44 | 0,37 | 0,363 | 0,14 | 1,46 |
| 3. | 0,44—0,6 | 0,52 | 0,514 | 0,16 | 1,36 |
| 4. | 0,6 —0,8 | 0,7 | 0,693 | 0,2 | 1,33 |
| 5. | 0,8 —1,05 | 0,925 | 0,917 | 0,25 | 1,31 |
| 6. | 1,05—1,39 | 1,22 | 1,208 | 0,34 | 1,32 |
| 7. | 1,39—1,85 | 1,62 | 1,604 | 0,46 | 1,33 |
| 8. | 1,85—2,51 | 2,18 | 2,155 | 0,66 | 1,35 |
| 9. | 2,51—3,45 | 1,98 | 2,943 | 0,94 | 1,37 |
| 10. | 3,45—4,85 | 4,15 | 4,091 | 1,4 | 1,4 |



DR. GORDOS GÉZA

1960-ban villamosmérnök, 1966-ban egyetemi doktori, 1977-ben kandidátusi oklevelet szerzett.

Fő munkahelye 1960-tól a BME Híradástechnikai Elektronika Intézete, ill. annak jogelődje, ahol jelenleg az átvitel- és rendszertechnika osztályt vezeti. 1964 és 1972 között a Posta Kísérleti Intézetben, 1972-ben UNESOO-szakértőként Görögországban, 1974/75-ben vendégprofesszorként Angliában dolgozott. Fő érdeklődési területe a fém- és fényvezetés digitális átvitel, adatátvitel, valamint a gépi beszédszintézis és beszédfelismerés.

$$H_N(Z^{-1}) =$$

$$C_0$$

$$= \frac{C_0}{(1 + 2A_1Z^{-1} + A_2Z^{-2})(1 + 2 \cdot B_1Z^{-1} + B_2Z^{-2})}$$

alakú, s ez a felírás egyúttal utal a szűrési funkció szoftver megvalósításának folyamatára is. A szerepiő együtthatók programozhatók, így a sávszűrők viszonylag kis munka árán átalakíthatók. A két másodfokú kaszkád alaptag „összekapcsolási” pontjában lehetőség van a jel bizonyos mértékű „erősítésére” (5i). Ez az erősítés csak 2 hatványa lehet, de már ez is kielégítő lehetőséget ad a digitális szűrők skálázására. A második kaszkád alaptag kimenetén szintén van egy skálázási lehetőség (7i az 1. ábrán).

Az i -edik szűrő kimeneti jelét egyenirányítjuk (8i), majd egy ablakintegrátorral (9i) az egyenirányított jel átlagértékét képezzük. Az ablak szélessége 150 minta, azaz 15 ms. Az átlagot a továbbiakban „sávenségiának” nevezzük.

A lényegkiemelés eredményét a 15 milliszekundumonként képezett 10 darab átlagérték, továbbá a 15 milliszekundum széles ablakban bekövetkezett nullátmetszések száma adja. Ez utóbbi érték a transzverzális szűréssel párhuzamosan képződik (10). Megjegyezzük, hogy nullátmetszésnek csak az olyan előjelváltásokat tekintjük, amelynél a közrefogó minták szorzata egy — beállítható — küszöbszámnál nagyobb, ezzel véde-

kezünk a jelszüneti zaj okozta téves számlálás ellen.

A host gép végezte szókezdet/szóvég detekció megkönnyítése érdekében a (11) blokk a 10 db átlagérték összegét képezi, 15 milliszekundumonként ez az adat is átvitelre kerül a jelfeldolgozó kártyáról a host-gépbe. Ezt nevezzük a szegmensösszenergiájának.

3.2. Kezdet/vég detekció

A helyes felismerés szempontjából nagy a jelentősége a szókezdet és a szóvég detektálásának. Mivel az akusztikus bevételezés a rendszerben folyamatos, ezért a kezdet/vég detekciónak is folyamatosan „gördülő” módon kell végbemennie. Az algoritmus a következő:

A TMS 32010 jelfeldolgozó processzor minden 15 msec-os időszegmens végén elhelyezi az adott szegmenshez tartozó lényegkiemelt jellemzőket a host gépben egy meghatározott memóriaterületen. Ezek az adatok: 10 sávenségi, a nullátmenetszám és az összenergia (a sávenségiák összege). A host gépen futó (PASCAL) program ezeket az adatokat gyűjti egy tömbben, amelyben gördülő módon mindig a legutóbbi 120 szegmens adatait tárolja. A „szókezdet/szóvég” meghatározást a program egy „kezdet/vég” detekcióból származtatja.

„Kezdetet” akkor detektál, ha olyan szegmens érkezik, amelynek összenergiája legalább 6 decibellel meghaladja az átlagos zajszintet.

„Véget” akkor detektál, ha egy korábban elkezdődött szó még nem fejeződött be és az összenergia 20 szegmenszen keresztül (a zajszint + 3) decibel alá esik.

A „kezdet” és a „vég” pozíciójából a program megállapítja a szó hosszát. Ha ez túlságosan rövid vagy túl hosszú, akkor nem tekintni valódi szónak és tovább folytatja a folyamatos akusztikus bevételezést, újabb szókezdetre várva. Ellenkező esetben a szókezdet a „kezdet” előtt, míg a szóvég a „vég” után van 90 ill. 105 msec-nyi védőközzel.

Ha a szó hossza olyan, hogy ennek alapján nem utasítható el (a jelenlegi beállításban 8-tól 80 szegmensig), akkor az akusztikus bevételezést leállítja és a szóhoz tartozó adatokat — a két végén megfelelő védőközzel ellátva — elhelyezi egy tömbben úgy, hogy a szó kezdete a tömb



KOUTNY ILONA

Koutny Ilona 1977-ben végzett az ELTE-n matematika-, francia és eszperantó szakon. Az ezt követő 5 évben számítástechnikával foglalkozott, (ebből 2 évig az SZKI-ban beszéd- és beszé-

felismeréssel) utána nyelvi majd tudományos anyagok szerkesztőjeként dolgozott. 1986-tól a BME Híradástechnikai Elektronika Intézetében beszédfelismeréssel foglalkozik. Fő érdeklődési területe a számítógépes nyelvészet.

elején legyen. Ez az adattömb kerül azután további feldolgozásra.

A TMS 32010 processzorral való kapcsolattartás és a működés szinkronizálása úgy történik, hogy a host gép (azaz a PASCAL program) bizonyos memóriaportokon keresztül adhat jelzéseket a jelfeldolgozó processzornak, az adatokat pedig egy közös memóriaterületen tudják átadni egymásnak. Ezek a műveletek TURBO—PASCAL-ból kényelmesen megvalósíthatók.

3.3. Energianormálás

A különböző hangosságú bemondások, valamint a száj és a mikrofon távolságának bizonytalan volta miatt szükséges a kapott energiaértékek nagyságának valamilyen normalizálása. Ez — a jelenlegi beállításban — úgy történik, hogy a maximális szegmens-energia éppen egy előre rögzített konstans legyen.

3.4. Időnormálás

A szavak közti távolság kiszámítása lényegesen egyszerűbb, ha az egyes szavakat reprezentáló vektorok egyforma hosszúak, azaz a szegmensszám



MAGYAR GÁBOR

A[†] Budapesti Műszaki Egyetem Villamosmérnöki Karán végzett 1981-ben. Azóta a BME HEI-ben különböző digitális jelfeldolgozási feladatokkal foglalkozott. Részt vett beszéddetektor, be-

szédszintézerek és beszédfelismerő létrehozásában. Tevékenysége elsősorban az ezeket megvalósító mikroprocesszoros és jelfeldolgozó rendszerek (hardver és gépközeli szoftver) kifejlesztésére illetve alkalmazására irányult.

azonos. Ezt az egyforma hossza alakítást nevezzük időnormálásnak. A feladatot úgy oldjuk meg, hogy ha a tekintett szó hosszabb, mint a normált hossz (amely a jelenlegi beállításban 40 szegmens), akkor kihagyunk az eredeti szóból annyi szegmenst, hogy éppen 40 maradjon. A kihagyandó szegmensek pozícióját egyenletesen osztjuk el az összes szegmens között.

Ha a szegmensek száma kisebb 40-nél, akkor hasonlóan járunk el, csak kihagyás helyett megduplázunk annyi szegmenst, hogy végül 40-et kapjunk.

3.5. Dinamikus idővetemítésen alapuló távolságszámítás

Az idővetemítés feladata az, hogy a különböző kiejtések közti sebességingadozásokat optimálisan kiegyenlítse. Ezek az ejtésritmus-ingadozások még akkor is megnehezítik a szavak összehasonlítását, ha egyébként a hosszuk már azonosra lett normálva.

Az idővetemítés problémakörének általános áttekintését itt megelőzzük, ezzel kapcsolatban az irodalomra utalunk (ld. pl. GORDÓS—TAKÁCS



DR. OSVÁTH LÁSZLÓ

A Budapesti Műszaki Egyetem Villamosmérnöki Karán kapott villamosmérnöki oklevelet 1975-ben. 1976 óta a BME Híradástechnikai Elektronikai Intézetének dolgozója. Érdeklődési területei a digitális jelfeldolgozás és az adatátvitel.

[1]). A VERBIDENT—SD—2 rendszerben azt a módszert alkalmazzuk, amely a dinamikus programozás néven ismert eljárás segítségével keresi meg az optimális időillesztést. A szakirodalom szerint ez a legjobban bevált módszer, a neve dinamikus idővetemítés (Dynamic Time Warping). Megjegyzendő, hogy az eljárásnak számos változata létezik (ld. pl. ŠAKOE—CHIBA [2], KUHN—TOMASCHESKI [3], MYERS—RABINER—ROSENBERG [4]).

Az általunk megvalósított algoritmust a következőképpen foglalhatjuk össze: Jelölje x_i a felismerendő szó i -edik szegmensét leíró vektort, melynek komponensei: $x_{i,1}, x_{i,2}, \dots, x_{i,11}$. ($x_{i,1} \dots x_{i,10}$ a sávenergiák, $x_{i,11}$ a nullátmenetszám, mégpedig alkalmas súlyozás után). Hasonlóképpen, jelölje y_j az éppen vizsgált szótárelem j -edik szegmensének vektorát, $y_{j,1}, y_{j,2}, \dots, y_{j,n}$ komponensekkel. Az x_i és y_j szegmensvektorok lokális távolságát az alábbi formulával definiáljuk:

$$d_{ij} = \sum_{k=1}^{11} |x_{ik} - y_{jk}|$$

A lokális távolságok felhasználásával kiszámítjuk az alábbi rekurzív formula alapján az ún. kumulatív távolságokat:



TAKÁCS GYÖRGY

Takács György okl. villamosmérnök a Posta Kísérleti Intézet Kapcsolástechnikai Osztályán dolgozik 1972 óta. A szintetikus beszéd alapuló távbeszélő tájékoztató szolgáltatások témáját irányítja. 1981 óta mellékfoglalkozásban a BME-HEI adjunktusa. A Magyar Tudományos Akadémia Akusztikai Komplex Bizottságának tagja.

$$t_{ij} = \min \begin{cases} t_{i-2, j-1} + 2t_{i, j-1} + d_{ij} \\ 2t_{i-1, j-1} + d_{ij} \\ t_{i-1, j-1} + 2t_{i-1, j} + d_{ij} \end{cases}$$

A kumulatív távolságokat csak az $|i-j| \leq 4$ egyenlőtlenséggel kijelölt megengedett tartományba eső (i, j) párokra számoljuk. (A tartomány határán a formulát értelemszerű módosítással használjuk, kihagyva a nemlétező ill. nem megengedett indexeket tartalmazó részeket.)

Végül, a $t_{40,40}$ kumulatív távolság adja az éppen összehasonlított két szó optimális időillesztéssel nyert távolságát. Ily módon kiszámítjuk a felismerendő szó „vetemített távolságát” az összes szótárelemtől és az így nyert távolságértékeket továbbítjuk az osztályozó algoritmus számára. A dinamikus idővetemítés tehát a távolságszámítással közös algoritmusban valósul meg.

Érdeemes még kiemelni a fent leírt algoritmust megvalósító szubrutin néhány további sajátosságát:

- A lokális távolságokat a TMS 32010 processzor számítja, a kumulált távolságokat pedig a host gépen futó PASCAL program. E két számítás párhuzamosan fut, kihasználva, hogy a t_{ij} kumulált távolsághoz még nincs szükség a d_{ki} lokális távolságra, ha $k > i$. Így amíg a PASCAL program számolja a kumulált távolságokat, addig a DSP a később szükséges lokális távolságokat készíti elő.
- A futási idő további csökkentése érdekében elkerültük az indexes változók használatát. Emiatt a szubrutin kevésbé „szép” és áttekinthető, viszont gyorsabb, ami a valós idejű működés szempontjából kritikus.
- Ha valamilyen i indexre minden t_{ij} kumulált távolság már eléri a bemondott szó valamely korábban már sorra vett szótárelemtől mért távolságát, akkor a számítást abbahagyjuk, hiszen ez a szótárelem már úgysem lehet legközelebbi szomszéd, mivel a kumulatív távolságok a számítás során csak növekedhetnek. Ez a beavatkozás mintegy 40–50%-kal csökkentette az átlagos futási időt.

3.6. Osztályozási algoritmus

Az osztályozás az előző pontban leírt idővetemített távolságok alapján történik, a legközelebbi szomszéd döntési eljárással, kibővítve azt még az előzőekkel:

a) Előszelekció a szó hossza alapján:

Ha a felismerendő szó és az éppen vizsgált szótárelem eredeti (nem normált) hossza kétszeres faktornál jobban különbözik, akkor azt a szótárelemet eleve elvetjük, így tőle már ki sem számoljuk a vetemített távolságot. Ezzel futási időt takarítunk meg.

b) Elutasítási küszöb:

Ha a minimális távolság nagyobb egy kísérleti-leg beállított küszöbnél, vagy az a) pont szerinti előszelekció minden szótárelemtet elutasított, akkor a felismerő ismételt bemondást kér.

c) Bizonytalansági küszöb:

Ha a legközelebbi szomszéd és a második legközelebbi szomszéd távolsága 10%-nál kevésbé különbözik és a két szó karakteres alakja nem egyezik meg, akkor a döntést bizonytalannak tekintjük és emiatt a felismerő ismétlést kér.

4. A VERBIDENT—SD—2 értékelése

A leírt algoritmussal gyors és biztonságos felismerés valósítható meg. A sebességről már említést tettünk (50...70 msec/szótárelem). Ez a könnyebben mérhető jellemző. A felismerés biztonságát a helyesen felismert bemondások százalékarányával jellemezhetjük. Csakhogy ez erősen függhet a szótár tartalmától, a bemondó hangjától, az akusztikai környezettől; egyszóval eléggé nehéz pontosan megragadni és egyértelmű mérési utasítások keretei közé szorítani. Így a kereskedelmi forgalomban levő rendszerekről az irodalomban ill. a cégek által közölt adatok is elég bizonytalannak.

A VERBIDENT—SD—2-n végzett mérések azt mutatják, hogy zajszegény környezetben, hibátlan szótárral a felismerés biztonsága legalább 95%, de több mérés eredményezett már 100%-os biztonságot is. Ez erősen függ a szótár konkrét összeállításától, vagyis attól, hogy tartalmaz-e akusztikailag nagyon hasonló szavakat.

Befejezésül megjegyezzük, hogy a fent leírt rendszernek még vannak további tartalékai is, mind a felismerési biztonság, mind pedig a reakcióidő javítása terén. Utóbbival kapcsolatban utalunk azokra az algoritmusokra, amelyeket a legközelebbi szomszéd keresés gyorsítására kerültek kidolgozásra. Ezek — általánosabb, nemcsak a beszédfelismeréshez kötődő jellegük miatt — máshol kerülnek részletes kifejezésre (ld. FARAGÓ—LINDER—LUGOSI—PIKLER [5], FARAGÓ [6]).

IRODALOM

- [1] Gordos G.—Takács Gy.: „Digitális beszédfeldolgozás”, Műszaki Könyvkiadó, Bp., 1983.
- [2] H. Sakoe—S. Chiba: „Dynamic programming algorithm optimization for spoken word recognition”, IEEE Trans. Acoust., Speech, Signal Process. Vol. ASSP—26, 1978.

- [3] *M. H. Kuhn—H. H. Tomaszewski*: „Improvements in isolated word recognition”, *IEEE Trans. Acoust., Speech, Signal Process.* vol. ASSP—31, Feb. 1983.
- [4] *C. Myers—L. H. Rabiner—A. E. Rosenberg*: „Performance Tradeoffs in dynamic time warping algorithms for isolated word recognition”, *IEEE Trans. Acoust., Speech, Signal Process.* vol. ASSP—28, Dec. 1980.
- [5] *Faragó A.—Linder T.—Lugosi G.—Pikler T.*: A legközelebbi szomszéd osztályozási módszer algoritmikus problémáiról, *Híradástechnika* (megjelenés alatt).
- [6] *Faragó A.*: On a combinatorial clustering problem. *Proceedings of the Conference of Program Designers (PD'87) ELTE—TTK, Bp. 1987.*
-