

## **1.Tartalomjegyzék**

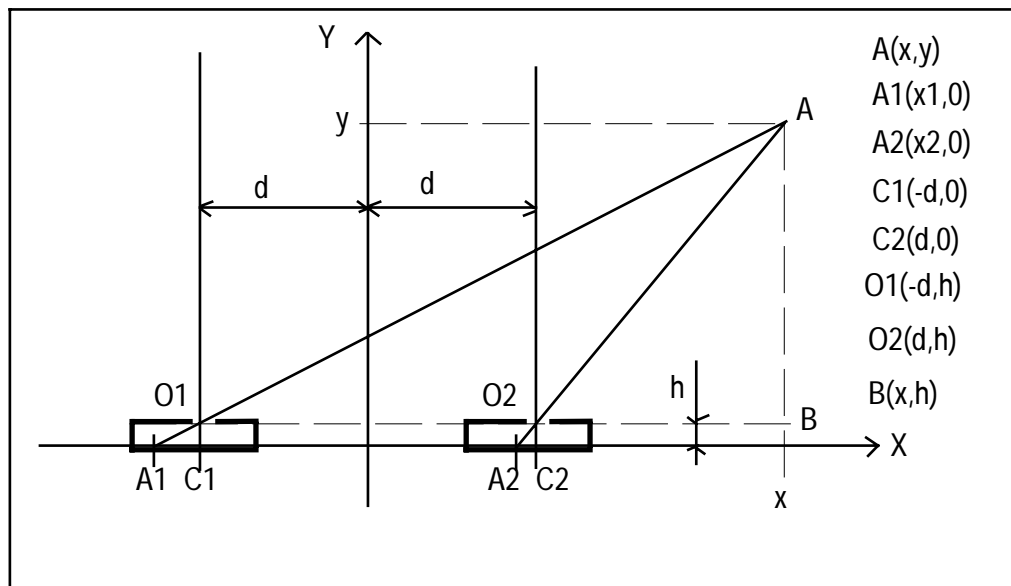
1.Tartalomjegyzék .....	1
2.Bevezetés .....	2
3.A matematikai modell kiválasztása .....	5
4.A vékony lap modell.....	7
5.Egy más módszer a matematikai modell kiválasztására.....	10
6.A felületet simaságát jellemző függvény diszkretizálása .....	14
7.Az interpolációt megvalósító neurális hálózat.....	17
7.1.A gradiens módszer .....	17
7.2.A template-ek megtervezése.....	22
7.3.Template tervezés más módszerekkel .....	30
7.3.1.Template tervezés a biharmonikus egyenlettel.....	30
7.3.2.Template tervezés a neurális hálózat energiaképletével .....	32
7.4.Az áramkör működése néhány példa kapcsán .....	33
8.Az approximációt megvalósító neurális hálózat.....	45
8.1.A gradiens módszer .....	46
8.2.A template-ek megtervezése.....	48
8.3.Az áramkör működése néhány példa kapcsán .....	50
9.Összefoglalás .....	54
10.Summary(összefoglalás angol nyelven) .....	56
11.Függelék.....	58
11.1.A celluláris neurális hálózat(CNN) .....	58
11.2.A szimulátorprogram .....	63
12.Irodalomjegyzék .....	66

## 2.Bevezetés

Sok területen szükség van valós tárgyak, alakzatok háromdimenziós képének ismeretére a tárgyak mechanikai érintése nélkül. Számos lehetséges módszer közül kézenfekvő az emberi látásból tanulnunk. A célszerűségi ok mellett az sem hanyagolandó el, hogy az emberi látáshoz kapcsolódó agyműködés megértése is serkenti az ezirányú kutatásokat. Ugyanis annak ellenére, hogy az agyról szerzett ismereteink igen hiányosak, a látórendszer egyes részei (amelyek a képfeldolgozás korai szakaszán találhatóak) viszonylag ismertek.

Tehát az emberi térlátástól tanulunk. Habár háromdimenziós világban élünk, az ember szemében kétdimenziós kép keletkezik. Mégis képesek vagyunk a tárgyakat felismerni, térbeli elrendezésüket megérteni. Vagyis az ember a két szemében keletkező kétdimenziós képből képes egy alakzat háromdimenziós képének rekonstrukciójára. Ez a sztereolátás.

Vizsgáljuk meg a térlátás működését egy egyszerű modell segítségével. Legyen a leképző eszköz két egyforma, fényt át nem eresztő falú doboz, egyik oldalán kis lyukkal. A doboz lyukkal szemközti oldalán keletkezik a kép. A következő ábra azt az esetet mutatja, amikor az egymás mellett elhelyezkedő dobozokkal az A pontról alkotunk képet.



2.1.Ábra

$A_1, A_2$  a két képpont.  $O_1, O_2$  a dobozok falán elhelyezkedő rések. A  $C_1, C_2$  pont az  $O_1, O_2$  merőleges vetülete az  $x$ -z síkra.

Az rögtön látható, hogy  $A_1$ ,  $A_2$  vetületi képpontok ismeretében  $A$  térbeli helye meghatározható, hiszen  $A$  az  $A_1O_1$  és az  $A_2O_2$  egyenesek metszéspontja.

Az elrendezés további analizéséhez egyszerűsítéssel élünk. Vizsgáljunk a kérdést síkproblémaként, vagyis legyen  $A$  az  $x$  és  $y$  tengelyek által meghatározott síkban. Az  $O_1$ ,  $O_2$  pontok  $y$  koordinátája  $h$ , a kép az  $x$  koordinátára képződik. Az  $O_1$ ,  $O_2$  pontok távolsága az  $y$  tengelytől  $d$ .  $B$  egy segédpont, amelyre később még szükség lesz. Az egyes pontok koordinátáit a kép mellett megadtuk.

Az  $O_1AB$  és az  $A_1O_1C_1$  derékszögű háromszög, illetve az  $O_2AB$  és az  $A_2O_2C_2$  derékszögű háromszög hasonló. Így a befogók aránya egyenlő:

$$\frac{y-h}{x+d} = \frac{h}{A_1C_1} \quad (2.1)$$

$$\frac{y-h}{x-d} = \frac{h}{A_2C_2}$$

A két egyenletet  $A_1C_1$ -re, illetve  $A_2C_2$ -re rendezve és egymásból kivonva, majd ismét rendezve:

$$y = \frac{2hd}{A_1C_1 - A_2C_2} + h \quad (2.2)$$

Vagyis ha ismerjük az  $A$  pont helyzetének eltérését az egyik képen a másik képhez képest (ez  $A_1C_1 - A_2C_2$ ), abból az  $A$  pont távolsága, vagyis  $y$ , egyszerűen számolható.

Tehát az alakzat háromdimenziós képének rekonstruálásakor az első lépés az, hogy elkészítjük a két vetületi képet. A két vetületi képen igyekszünk minél több egymásnak megfelelő képpontpárt találni (amelyek egy tárgyponthoz tartoznak). Így az alakzat számos pontjának térbeli helyét ismerjük, bár némely pont helyzetét a valóságban működő algoritmusok rosszul határozzák meg.

Eddig csak egyes pontok térbeli helyét ismerjük, végeredményként azonban a tárgy felületét várjuk. Ezért a tárgy térbeli képének rekonstrukciójakor a második elemi feladat ezekre az ismert pontokra egy felületet illeszteni. Diplomamunkám ezzel az illesztési problémával foglalkozik. De mindezt nem számítógépes programmal valósítom meg, hanem celluláris neurális hálózattal. (A celluláris neurális hálózatról bővebb ismeretek találhatóak a függelékben.)

Az illesztés két alapvető módszerét vizsgáltam. Az egyszerűbb esetben a felületnek pontosan át kell haladnia az ismert pontokon. Ezt nevezzük *interpolációnak*. Végtelen olyan felület van, amely áthalad az ismert pontokon. Ezek közül kell egyet, mint optimálisat kiválasztani. Nyilván jó, ha ez a felület minél simább. Ennek a tulajdonságnak kellett matematikai megfogalmazását megtalálnom.

Az előző módszer nagy hátránya, hogy egy rosszul megadott pont az egész felületet eltorzíthatja, ugyanis a megadott pontokon a felületet feltétlenül átvezeti. Ezen segít az illesztés egy bonyolultabb esete, amikor az illesztett felületnek nem kell pontosan áthaladnia az ismert pontokon. Ezt *approximációnak* hívjuk.

Diplomamunkám első részében a matematikai modellválasztás problémáját tekintem át. Majd külön-külön az előbbi két módszer esetén tárgyalom a megvalósítandó algoritmust, annak alkalmazását celluláris neurális hálózatra. A tervezett áramkör működését konkrét példák kapcsán mutatom be.

### **3.A matematikai modell kiválasztása**

A feladatunk ismert térbeli pontokra egy háromdimenziós felületet illeszteni. Nyilván számtalan ilyen felület lehetséges, ezért meg kell határozni egy olyan matematikai kritériumot, amely alapján e felületek közül egy, mint optimális, kiválasztható. Törekedni kell arra, hogy az ismert pontokra minél simább felületet illesszünk. Ennek a simaságnak kell megtalálni matematikai megfogalmazását. Ehhez fizikai megfontolásokat is fel fogunk használni.

Tekintsük először az egydimenziós esetet. Terzopoulos [4] a klasszikus ún. spline interpolációs módszerből indult ki, amely a

$$v(x_i) = c_i, \quad 1 \leq i \leq N_c \quad (3.1)$$

kényszerfeltételnek megfelelő,  $[a,b]$ intervallumon értelmezett  $v(x)$  függvényekből azt választja ki, amelyre a

$$|v|_m^2 = \int_a^b \left| \frac{d^m v(x)}{dx^m} \right|^2 dx, \quad N_c \geq m \geq 1 \quad (3.2)$$

norma minimális. Az  $x_i$  azoknak a megadott pontoknak az  $x$  koordinátái, amelyen a függvénygörbének mindenképpen át kell haladnia, vagyis  $x_i$  helyen  $c_i$  értéket kell a függvénynek felvennie.  $N_c$  a megadott pontok száma.

Nyilván az integrálás elvégzéséhez az  $m$ . deriválnak  $[a,b]$  minden pontjában léteznie kell. Ennek szükséges feltétele az  $m-1$ . derivált folytonossága.

Kérdés, hogy az  $m$  értéket mekkorára válasszuk a konkrét esetben. Itt fizikai megfontolásokat kell segítségül hívnunk.

Az  $m=1$  esetben a norma alakja:

$$|v|_1^2 = \int_a^b \left( \frac{dv}{dx} \right)^2 dx \quad (3.3)$$

Ilyen alakú a végtelen hosszú ideális fonál potenciális energiáját megadó képlet. Egy mechanikai rendszer olyan egyensúlyi helyzetet vesz fel, amelyben adott kényszerek mellett potenciális energiája minimális. Jelen esetben a fonál úgy áll be, hogy két rögzítési pont között egyenesen halad. Vagyis ha azt a függvényt választjuk ki, amelyre ez a norma minimális, lineáris interpolációt kapunk. Ismert tény, hogy ez a minimum kielégíti a

$$\frac{\partial^2 v}{\partial x^2} = 0 \quad (3.4)$$

egyenletet. (Ez varációszámítással bizonyítható. Lásd Terzopoulos [4].)

Az  $m=2$  esetben a norma alakja:

$$|v|_2^2 = \int_a^b \left( \frac{d^2 v}{dx^2} \right)^2 dx \quad (3.5)$$

Ilyen alakú a végtelen hosszú rugalmas rúd rugalmassági energiáját megadó képlet hajlítás esetén. Az előbbi függvény minimuma a következő egyenletet elégíti ki:

$$\frac{\partial^4 v}{\partial x^4} = 0 \quad (3.6)$$

Kétdimenziós esetben a fonal megfelelője a membrán vagy hártya, a rúd pedig a vékony, rugalmas lap.

A membránok ( $m=1$ ) által generált felületek nem elég simák. Hasonlóképpen, mint ahogy egydimenziós megfelelőjük által nyújtott lineáris közelítés sem az.

A vékony lap ( $m=2$ ) nyújtotta lehetőségek úgy tűnik, hogy megfelelőek.

A matematikai megfontolások mellett a sztereolátásról szerzett ismeretek is az  $m=2$  esetet támogatják. Ezért a szakirodalom, nevezetesen Grimson [1] és Terzopoulos [4] is ezt a változatot részesítette előnyben. A továbbiakban tehát a vékony lapot fogjuk modellként alkalmazni.

#### **4.A vékony lap modell**

Fizikai modellnek a vékony lapot választottuk. Ez azt jelenti, hogy az általunk választott módszer olyan felületet eredményez, mint ahogy adott rögzítések mellett a vékony lap beáll. Fizikai rendszerek potenciális energiája adott kényszerek mellett egyensúlyi állapotban minimális. Ezért meghatározzuk a vékony lap potenciális energiáját. A kívánt felület az lesz, amelyre ez minimális.

Legyen  $v(x,y)$  a felületet leíró függvény, amely megadja a felület adott pontjának alapsíktól mért magasságát. Így tehát a felületet most a  $v(x,y)$  függvény reprezentálja.

Az interpolációs probléma kedvéért olyan elrendezést állítottunk össze, amely később következtetések levonását teszi lehetővé. Így a vékony laphoz egyes pontokon rugók csatlakoznak. Ezek testesítik meg azt az elvárásunkat, hogy adott  $(x_i, y_i)$  koordinátákkal jellemzett pontokban a felület az ismert pontokhoz minél közelebb haladjon el.

A vékony lap alapsíktól mért magassága az  $(x_i, y_i)$  pontban  $v(x_i, y_i)$ . A felület ezen pontjához rugó csatlakozik, a rugók vége nyújtatlan állapotban a  $c_{(x_i, y_i)}$  magasságú pontban lenne. Nyilván a rugó arra törekszik, hogy adott pontban a felület  $v(x_i, y_i)$  magassát  $c_{(x_i, y_i)}$ -hoz közelítse.

Adott körülmények közt a rendszer úgy áll be, hogy a potenciális energiák  $E(v(x,y))$ összege a minimális legyen. Az  $E$  energia két tag összege, a vékony lapban felhalmozódott energiáé és a rugók rugalmassági energiájé.

$$E(v(x, y)) = E_{\text{vékony lap}} + E_{\text{rugók}} \quad (4.1)$$

A vékony lapra vonatkozó energiaképlet Hilbert [5] szerint a következőképp alakul:

$$\begin{aligned}
E_{\text{vékony lap}} = & \iint_{\Omega} \frac{1}{2} (\Delta v)^2 - (1 - \sigma) \left( \frac{\partial^2 v}{\partial x^2} \frac{\partial^2 v}{\partial y^2} - \left( \frac{\partial^2 v}{\partial x \partial y} \right)^2 \right) dx dy - \\
& \iint_{\Omega} g v dx dy - \\
& \iint_{\partial \Omega} p(s) v ds - \\
& \iint_{\partial \Omega} m(s) \frac{\partial v}{\partial n} ds +
\end{aligned} \tag{4.2}$$

Az első tag az  $\Omega$  felülettel jellemzett lapban ébredő feszültségekkel kapcsolatos energiát jelenti. A  $\sigma$  a Poisson-féle szám, más néven harántösszehúzóási együttható. A megnyúlt anyag átmérője csökken. Ezt a hatást jellemzi a Poisson-féle szám.

A második tag a  $g(x,y)$  függvénnyel reprezentált, függőlegesen lefele mutató lapra ható erő hatását mutatja. Ilyen pl. a gravitáció. Tehát ez a tag tulajdonképpen a magassági energia.

A harmadik és negyedik tag a lap  $d\Omega$  kerületére ható  $p(s)$ -sel jellemzett külső erőt és a kerületen jelentkező  $m(s)$ -sel jellemzett hajlítónyomatékot jelenti. Az  $n$  szerinti parciális iránymenti deriváltat jelöl. Az irány itt a felület határának kifelé mutató normálisa. Az  $s$  ívhossz a  $d\Omega$  kerület mentén.

A második energiateg pedig a  $\Delta x$ -szel megnyúlt rugóra vonatkozó

$$E_{\text{potenciális}} = \frac{1}{2} \alpha (\Delta x)^2 \tag{4.3}$$

potenciális energiát megadó képletből adódóan:

$$E_{\text{rugók}} = \frac{1}{2} \sum_{(x_i, y_i) \in I} \alpha (v(x_i, y_i) - c_{(x_i, y_i)})^2 \tag{4.4}$$

Itt  $x_i, y_i, c_{(x_i, y_i)}$  az  $i$ . rugó végének három koordinátája nyújtatlan állapotban,  $I$  az ismert pontok  $(x_i, y_i)$  koordinátapárjainak halmaza,  $\alpha$  a rugók direkciós ereje.

Előállt tehát az energiaképlet, azonban szükséges bizonyos egyszerűsítésekkel élni. Esetünkben sem a gravitációs erő nem hat, sem a peremen jelentkező erők. Így a lap potenciális energiáját megadó tagok közül csupán az első marad. A  $\sigma$  konstans 0-nak választva a teljes potenciális energiára ez adódik:

$$E_{teljes} = \iint_{\Omega} \frac{1}{2} (\Delta v)^2 - \left( \frac{\partial^2 v}{\partial x^2} \frac{\partial^2 v}{\partial y^2} - \left( \frac{\partial^2 v}{\partial x \partial y} \right)^2 \right) dx dy + \frac{1}{2} \sum_{(x_i, y_i) \in I} \alpha [v(x_i, y_i) - c_{(x_i, y_i)}]^2 \quad (4.5)$$

Felhasználva, hogy

$$\Delta v = \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \quad (4.6)$$

a képlet tovább alakítható:

$$E_{teljes} = \frac{1}{2} \iint \left( \frac{\partial^2 v}{\partial x^2} \right)^2 + 2 \left( \frac{\partial^2 v}{\partial x \partial y} \right)^2 + \left( \frac{\partial^2 v}{\partial y^2} \right)^2 dx dy + \frac{1}{2} \sum_{(x_i, y_i) \in I} \alpha [v(x_i, y_i) - c_{(x_i, y_i)}]^2 \quad (4.7)$$

A két tag szemléletes jelentéssel bír. Az első annál kisebb, minél simább a függvény. Vagyis a  $v(x,y)$ -nal megadott felület simaságát jellemzi. A második annál kisebb, minél jobban illeszkedik a felület az ismert pontokra.

A továbbiakban az 1/2-es szorzót elhagyjuk, hiszen a minimum így nem változik, a számolás viszont egyszerűsödik.

## **5.Egy más módszer a matematikai modell kiválasztására**

Grimson [3] más megfontolások után jutott ugyanarra az eredményre. Ő egy olyan függvényt keresett, amely jellemzi a felület simaságát. Adott felületre ennek az  $E(v)$  függvénynek az értéke annál kisebb, minél simább az a felület. Tehát a legsimább felületre az  $E$  függvény minimális. A következő feltételeket szabta a kiválasztandó függvényre:

1. Legyen a másodrendű deriváltak függvénye, mivel azt szeretnénk, ha a felület "irányváltásait" mérné. Az első deriváltak a felület meredekségére utalnak, a második deriváltak ennek a meredekségnek a megváltozására.

2. A függvény felírható legyen, mint

$$E(v) = \mu(v, v)^{\frac{1}{2}}, \quad (5.1)$$



ahol a  $\mu$ -vel jelölt skaláris szorzatra a következő tulajdonságok érvényesek:

$$\begin{aligned}
 1. & \mu(f, g) = \mu(g, f) \\
 2. & \mu(f + g, h) = \mu(f, h) + \mu(g, h) \\
 3. & \mu(\alpha f, g) = \alpha \mu(f, g) \\
 4. & \mu(f, f) \geq 0
 \end{aligned}
 \tag{5.2}$$

Ez az egyértelmű megoldás miatt szükséges. Grimson [3] bebizonyította, hogy ekkor létezik olyan felület, amelyre a függvény a többi felület közül a legkisebb értéket adja. Ez a minimum az  $E(v)$  nullterének elemeitől eltekintve egyértelmű. ( $E(v)$  nulltere azon  $v$  felületek halmaza, amelyre  $E=0$ .) Az, hogy az  $E$  függvény két különböző felületre is minimumot ad (vagyis  $E(v_1)=E(v_2)$ ) csak úgy lehetséges, hogy  $v_1(x,y)-v_2(x,y)$   $E$  nullterének eleme, vagyis  $E(v_1(x,y)-v_2(x,y))=0$ .

3.A függvényérték ne változzon, ha a felületet elforgatjuk.

A megoldások két függvény lineáris kombinációi. Ezek:

$$E_1(v(x, y)) = \left( \iint \left( \frac{\partial^2 v}{\partial x^2} \right) + 2 \left( \frac{\partial^2 v}{\partial x \partial y} \right) + \left( \frac{\partial^2 v}{\partial y^2} \right) dx dy \right)^{\frac{1}{2}}
 \tag{5.3}$$

$$E_2(v) = \left( \iint (\nabla^2 v)^2 dx dy \right)^{\frac{1}{2}}
 \tag{5.4}$$

$E_1$  nulltere (vagyis azon  $v$  felületek halmaza, amelyre 0-át ad) a lineáris függvények halmaza.  $E_2$  nulltere a harmonikus függvények halmaza (ennek  $E_1$  nulltere részhalmaza). Ez azt eredményezi, hogy az egymástól egy lineáris additív tagban különböző (minimumot adó)  $v$  felületeket  $E_1$  nem tudja megkülönböztetni.  $E_2$  pedig ugyan olyan értéket ad az egymástól egy harmonikusokból álló additív tagban különböző függvényre. A két függvény lineáris kombinációinak nulltere tartalmazza a lineáris függvényeket, és esetleg más függvényeket is. Nyilván célunk a minél kisebb nulltér, ezért  $E_1$  kedvezőbbnek látszik.

Vizsgáljuk meg, létezik-e az adott pontokon áthaladó két minimumot adó felület, amelyre  $E_1$ , illetve  $E_2$  egyenlő.  $E_1$  akkor adna két ilyen felületre azonos értéket, ha azok  $x$  és  $y$  egy lineáris függvényben különböznenek. Két pont esetén ez még elképzelhető, de három, nem egy egyenesbe eső pont esetén már nem. Ugyanakkor  $E_2$ -nél találhatunk két olyan függvényt, amelyek átmennek az ismert pontokon és csak harmonikusokban különböznek. Ez döntő

érv  $E_1$  mellett. Hiszen  $E_2$ -öt alkalmazva előfordulhatna, hogy ugyan azokra az ismert pontokra több minimumot adó felületet is tud illeszteni.

Variációszámítással belátható, hogy mindkét megoldás kielégíti a

$$\nabla^4 v = \frac{\partial^4 v}{\partial x^4} + 2 \frac{\partial^4 v}{\partial x^2 \partial y^2} + \frac{\partial^4 v}{\partial y^4} = 0 \quad (5.5)$$

biharmonikus egyenletet, csupán más peremfeltételekkel. Ebből sejthetjük, hogy a kapott megoldások leginkább a szélek mentén különböznek. Ez így is van. Grimson [3] mindkét módszert kipróbálta felületinterpolációra. A  $E_2$ -t minimalizáló a széleken torzulásokhoz vezetett. Általános értelemben az eredmény nem volt elég sima. Ez is hozzájárult, hogy Grimson az első függvényt választotta. Így a felület simaságát mérő függvény a gyökvonás elhagyásával (mivel ez a minimumot adó  $v(x,y)$  függvényen nem változtat, de a számolást könnyíti):

$$E(v(x, y)) = \iint \left( \frac{\partial^2 v}{\partial x^2} \right)^2 + 2 \left( \frac{\partial^2 v}{\partial x \partial y} \right)^2 + \left( \frac{\partial^2 v}{\partial y^2} \right)^2 dx dy \quad (5.6)$$

Interpoláció esetén, ha a felületnek mindenképpen át kell haladnia az ismert pontokon, azt a felületet kell kiválasztani, amelyre ez a kifejezés minimális. A másik eset az approximáció. (Lásd Grimson [1].) Ilyenkor az ismert pontokon nem kell pontosan átmennie a felületnek. Ekkor a felületet értékelő függvényben megjelenik egy második tag is, amely az illeszkedés pontosságát méri.

$$E(v(x, y)) = \iint \left( \frac{\partial^2 v}{\partial x^2} \right)^2 + 2 \left( \frac{\partial^2 v}{\partial x \partial y} \right)^2 + \left( \frac{\partial^2 v}{\partial y^2} \right)^2 dx dy + \sum_{(x_i, y_i) \in I} \alpha [v(x_i, y_i) - c_{(x_i, y_i)}]^2 \quad (5.7)$$

Tehát ez egy olyan függvény, amely annál kisebb, minél kedvezőbb számunkra a felület. Az első tag a simaságot méri, a második az illeszkedést.

Így Grimson is hasonló eredményre jutott, bár más megfontolások után.

**6.A felületet simaságát jellemző függvény diszkretizálása**

A mi problémánk esetén a felületek egy síkrács segítségével vannak megadva. Minden egyes rácsponthoz egy valós szám tartozik, amely a felület adott pontjának a síktól mért magasságát mutatja. Tehát az  $x$  és  $y$  koordináta nem folytonos, hanem diszkrét. A következőkben kiszámítjuk a minimalizálandó energiafüggvényt erre a diszkrét esetre a Grimson [1] által javasolt módszer szerint.

Először a függvény első tagjával foglalkozunk. Alakja a következő:

$$E_{\text{vékony lap}}(v(x, y)) = \iint \left( \frac{\partial^2 v}{\partial x^2} \right)^2 + 2 \left( \frac{\partial^2 v}{\partial x \partial y} \right)^2 + \left( \frac{\partial^2 v}{\partial y^2} \right)^2 dx dy \quad (6.1)$$

Egy-egy rácspont helyét két egész szám jelöli ki. A továbbiakban az  $(i, j)$  koordinátákkal jellemzett rácselemnél a felület magasságát  $v_{(i, j)}$  jelöli. A teljes  $m^*m$  méretű rács jellemző  $v_{(i, j)}$  értékeiből egy vektort képezhetünk. Ez az  $m^*m$  elemű  $\underline{v}$  vektor a teljes felületet megadja.

$$\underline{v} = \begin{pmatrix} v_{(0,0)} \\ v_{(0,1)} \\ \dots \\ v_{(m-1, m-2)} \\ v_{(m-1, m-1)} \end{pmatrix} \quad (6.2)$$

Ezzel a vektorral az aktuális felületet úgy adjuk meg, mint a lehetséges felületeket tartalmazó  $m^*m$  dimenziós tér egy pontját.

A diszkrét formára hozás első lépése a differenciálhányadosok differenciahányadossá alakítása. A  $h$  a két rácspont közti távolság.

$$\frac{\partial^2 v_{(i, j)}}{\partial x^2} = \frac{1}{h^2} [v_{(i+1, j)} - 2v_{(i, j)} + v_{(i-1, j)}] + O(h^2) \quad (6.3)$$

$$\frac{\partial^2 v_{(i, j)}}{\partial y^2} = \frac{1}{h^2} [v_{(i, j+1)} - 2v_{(i, j)} + v_{(i, j-1)}] + O(h^2)$$

$$\frac{\partial^2 v_{(i, j)}}{\partial x \partial y} = \frac{1}{4h^2} [v_{(i+1, j+1)} - v_{(i+1, j-1)} - v_{(i-1, j+1)} + v_{(i-1, j-1)}] + O(h^2)$$

$O(h^2)$  a tényleges érték és a közelítésünk közti hiba. Ha  $h$  nullához tart, ez is nullához tart.

Ezek után következnek az integrálás diszkretizálása, vagyis az integrálás szummává alakítása. Így az  $E(\underline{v})$  függvény a következő alakot ölti:

$$\begin{aligned}
 E_{\text{vékony lap}}(\underline{v}) &= \sum_{i=1}^{m-2} \sum_{j=0}^{m-1} (v_{(i-1,j)} - 2v_{(i,j)} + v_{(i+1,j)})^2 + \\
 &\sum_{i=0}^{m-1} \sum_{j=1}^{m-2} (v_{(i,j-1)} - 2v_{(i,j)} + v_{(i,j+1)})^2 + \\
 &2 \sum_{i=0}^{m-2} \sum_{j=0}^{m-2} (v_{(i,j)} - v_{(i+1,j)} - v_{(i,j+1)} + v_{(i+1,j+1)})^2
 \end{aligned} \tag{6.4}$$

Ehhez járul az illeszkedést vizsgáló második tag. Legyen  $I$  azon rácspontok  $(i,j)$  koordinátáinak halmaza, amelyeknél a felület síktól mért magasságát ismerjük.

$$I = \{(i, j) \mid \text{ismerjük a magasságot az } (i, j) \text{ rácspontban}\} \tag{6.5}$$

A másik új jelölés  $c_{(i,j)}$ . Ez az ismert magasságérték az  $(i,j)$  pontban.

Ezekkel a második energiatag diszkretizált alakja:

$$E_{\text{rugók}}(\underline{v}) = \alpha \sum_{(i,j) \in I} (v_{(i,j)} - c_{(i,j)})^2 \tag{6.6}$$

Így a vékony lap potenciális energiájának mindkét tagot tartalmazó diszkretizált alakja:

$$\begin{aligned}
 E(\underline{v}) &= \sum_{i=1}^{m-2} \sum_{j=0}^{m-1} (v_{(i-1,j)} - 2v_{(i,j)} + v_{(i+1,j)})^2 + \\
 &\sum_{i=0}^{m-1} \sum_{j=1}^{m-2} (v_{(i,j-1)} - 2v_{(i,j)} + v_{(i,j+1)})^2 + \\
 &2 \sum_{i=0}^{m-2} \sum_{j=0}^{m-2} (v_{(i,j)} - v_{(i+1,j)} - v_{(i,j+1)} + v_{(i+1,j+1)})^2 + \\
 &\alpha \sum_{(i,j) \in I} (v_{(i,j)} - c_{(i,j)})^2
 \end{aligned} \tag{6.7}$$

## **7.Az interpolációt megvalósító neurális hálózat**

A probléma egyszerűsödik, ha az illesztett felületnek az ismert pontokon mindenképpen át kell haladnia. A vékony lap analógiában ez annak felel meg, amikor a vékony lapot az ismert pontokhoz mereven rögzítjük rugók nélkül. Ezt az esetet interpolációnak nevezzük. Ilyenkor a minimalizálandó függvény csupán a felület simaságát meghatározó első tagot tartalmazza.

### **7.1.A gradiens módszer**

A minimum megkeresésére az ún. gradiens módszert használjuk. Ez az eljárás azt használja ki, hogy a  $\underline{v}$  vektor által kijelölt pontból kis  $\Delta\underline{v}$ -vel elmozdulva az  $E(\underline{v})$  függvény értéke  $\text{grad}E(\underline{v})$  irányban nő a legtöbbet, illetve  $-\text{grad}E(\underline{v})$  irányban csökken a legtöbbet. Mi a függvény minimumát keressük, ezért a  $-\text{grad}E(\underline{v})$  irányban indulunk el. Ily módon a  $\underline{v}_0$  kiindulási pontból az első iteráció után a következőt kapjuk:

$$\underline{v}_1 = \underline{v}_0 - \beta \text{grad}E(\underline{v})|_{\underline{v}_0} \quad (7.1)$$

Általában a  $(k+1)$ . pontot a  $k$ . pontból:

$$\underline{v}_{k+1} = \underline{v}_k - \beta \text{grad}E(\underline{v})|_{\underline{v}_k} \quad (7.2)$$

A  $\beta$  állandót úgy választjuk meg, hogy  $-\text{grad}E(\underline{v})$  irányában haladva a következő iterációs pontban  $E(\underline{v})$  értéke a lehető legkisebb legyen.  $E(\underline{v})$  ismeretében  $\beta$ , mint  $\underline{v}$  függvénye meghatározható (Grimson [1]).

Tehát ez a módszer alkalmas az  $E(\underline{v})$  függvény lokális minimumhelyének megtalálására. (Bővebben lásd [10].) A főbb lépések jelen esetben a következők:

0. A  $\underline{v}_0$  kezdeti állapot meghatározása.

$$\forall (i, j) \in I: v_{0(i,j)} = c_{(i,j)}$$

$$\forall (i, j) \notin I: v_{0(i,j)} = 0$$

1.  $\text{grad}(E(\underline{v}))$  számítása a  $\underline{v}_k$  aktuális iterációs pontban

2.  $\beta$  számítása  $\underline{v}$  függvényében.

3. A következő iterációs pont számítása.

$$\underline{v}_{k+1} = \underline{v}_k - \beta \text{grad}E(\underline{v})|_{\underline{v}_k}$$

(Kivéve az ismert pontokat. Itt a felület  $c_{(i,j)}$  magassértékei nem változnak.)

4. Ha  $|\Delta \underline{v}| > \varepsilon$ , vissza az 1. lépésre.

(7.3)

Tehát a

$$E(\underline{v}) = \sum_{i=1}^{m-2} \sum_{j=0}^{m-1} (v_{(i-1,j)} - 2v_{(i,j)} + v_{(i+1,j)})^2 +$$

$$\sum_{i=0}^{m-1} \sum_{j=1}^{m-2} (v_{(i,j-1)} - 2v_{(i,j)} + v_{(i,j+1)})^2 +$$

$$2 \sum_{i=0}^{m-2} \sum_{j=0}^{m-2} (v_{(i,j)} - v_{(i+1,j)} - v_{(i,j+1)} + v_{(i+1,j+1)})^2$$
(7.4)

függvény minimumát keressük a

$$v_{(i,j)} - c_{(i,j)} = 0, \forall (i, j) \in I$$
(7.5)

feltétel mellett. (Utóbbi képlet annak az elvárásnak matematikai megfogalmazása, hogy a felület pontosan áthaladjon az ismert pontokon.)

Ha felbontjuk a zárójelket az  $E(\underline{v})$  függvény kifejezésében, azt tapasztaljuk hogy, háromtényezős szorzatok összegéből áll. Egy tényező egy konstans és két rácspont magasságértékeinek szorzata. Megfigyelhető, hogy a szorzatban szereplő magasságértékek olyan rácspontokhoz tartoznak, amelyek egymásnak  $r=2$  sugarú környezetében vannak.

$$E(\underline{v}) = \frac{1}{2} \sum_{(i,j)} \sum_{(k,l) \in N_r(i,j)} C(i,j,k,l) v_{(i,j)} v_{(k,l)}, \quad r=2 \quad (7.6)$$

$$C(i,j,k,l) = C(k,l,i,j)$$

A módszerhez szükséges a gradiens meghatározása. Ha az energiafüggvény az előbbi alakú, akkor a  $\text{grad}E(\underline{v})$  egy koordinátája a következőképp alakul:

$$\frac{\partial E}{\partial v_{(i,j)}} = \sum_{(k,l) \in N_r(i,j)} C(i,j,k,l) v_{(k,l)}, \quad r=2 \quad (7.7)$$

A kifejezést  $v_{(k,l)}$  szerint deriválva megkapjuk a C konstansokat is.

$$C(i,j,k,l) = \frac{\partial^2 E}{\partial v_{(i,j)} \partial v_{(k,l)}} \quad (7.8)$$

A  $C(i,j,k,l)$  konstansokat az energiafüggvény (7.4) alakjából kaphatjuk, úgy, hogy felbontjuk a zárójeleket és összehasonlítjuk a (7.6) kifejezéssel. A  $C(i,j,i+\Delta i,j+\Delta j)$  értékeket a következő táblázatok tartalmazzák. (Ha  $\Delta i$  vagy  $\Delta j$  abszolút értéke kettőnél nagyobb, C értéke nulla.) Nyilván ezek a táblázatok függenek  $i$ -től és  $j$ -től.

	$\Delta i / \Delta j$	-2	-1	0	1	2	
a.)	-2	0	0	2	0	0	(7.9)
	-1	0	4	-16	4	0	
	0	2	-16	40	-16	2	
	1	0	4	-16	4	0	
	2	0	0	2	0	0	

	$\Delta i / \Delta j$	-2	-1	0	1	2	
b.)	-2	0	0	0	0	0	(7.10)
	-1	0	4	-12	4	0	
	0	2	-16	38	-16	2	
	1	0	4	-16	4	0	
	2	0	0	2	0	0	

$$\begin{array}{c}
 \begin{array}{c|ccccc}
 \Delta i / \Delta j & -2 & -1 & 0 & 1 & 2 \\
 \hline
 -2 & 0 & 0 & 0 & 0 & 0 \\
 -1 & 0 & 0 & 0 & 0 & 0 \\
 0 & 2 & -12 & 22 & -12 & 2 \\
 1 & 0 & 4 & -12 & 4 & 0 \\
 2 & 0 & 0 & 2 & 0 & 0
 \end{array} \\
 \text{c.)}
 \end{array}
 \tag{7.11}$$

$$\begin{array}{c}
 \begin{array}{c|ccccc}
 \Delta i / \Delta j & -2 & -1 & 0 & 1 & 2 \\
 \hline
 -2 & 0 & 0 & 0 & 0 & 0 \\
 -1 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & -8 & 20 & -12 & 2 \\
 1 & 0 & 4 & -12 & 4 & 0 \\
 2 & 0 & 0 & 2 & 0 & 0
 \end{array} \\
 \text{d.)}
 \end{array}
 \tag{7.12}$$

$$\begin{array}{c}
 \begin{array}{c|ccccc}
 \Delta i / \Delta j & -2 & -1 & 0 & 1 & 2 \\
 \hline
 -2 & 0 & 0 & 0 & 0 & 0 \\
 -1 & 0 & 4 & -12 & 4 & 0 \\
 0 & 0 & -12 & 36 & -16 & 2 \\
 1 & 0 & 4 & -12 & 4 & 0 \\
 2 & 0 & 0 & 2 & 0 & 0
 \end{array} \\
 \text{e.)}
 \end{array}
 \tag{7.13}$$

$$\begin{array}{c}
 \begin{array}{c|ccccc}
 \Delta i / \Delta j & -2 & -1 & 0 & 1 & 2 \\
 \hline
 -2 & 0 & 0 & 0 & 0 & 0 \\
 -1 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 8 & -8 & 2 \\
 1 & 0 & 0 & -8 & 4 & 0 \\
 2 & 0 & 0 & 2 & 0 & 0
 \end{array} \\
 \text{f.)}
 \end{array}
 \tag{7.14}$$

A következő táblázat azt mutatja, adott  $i$  és  $j$  értékek mellett melyik kis táblázatot kell figyelembe venni. Az első betű a hat kis táblázat egyikére utal. "X", "y", "z" tükrözést jelent az  $x$ , az  $y$  tengelyre, illetve az  $x$  és  $y$  tengellyel egyaránt 45 fokos szöget bezáró egyenesre.



i / j	0	1	2	3	4	.	m - 3	m - 2	m - 1
0	f	d	c	c	c	.	c	dy	fy
1	dz	e	b	b	b	.	b	ey	dzy
2	cz	bz	a	a	a	.	a	bzy	czy
3	cz	bz	a	a	a	.	a	bzy	czy
4	cz	bz	a	a	a	.	a	bzy	czy
.	.	.	.	.	.	.	.	.	.
m - 3	cz	bz	a	a	a	.	a	bzy	czy
m - 2	dzx	ex	bx	bx	bx	.	bx	exy	dzxy
m - 1	fx	dx	cx	cx	cx	.	cx	dxy	fxxy

7.1. Táblázat

Tehát például ha  $m=9$ , az energiafüggvény deriváltja a rács középső pontjához tartozó magasságérték szerint (az "a" jelű kis táblázatot felhasználva):

$$\frac{\partial E}{\partial v_{(4,4)}} = 40v_{(4,4)} - 16(v_{(3,4)} + v_{(5,4)} + v_{(4,3)} + v_{(4,5)}) + 4(v_{(3,3)} + v_{(5,3)} + v_{(5,3)} + v_{(5,5)}) + 2(v_{(2,4)} + v_{(6,4)} + v_{(4,2)} + v_{(4,6)}) \quad (7.15)$$

Fontos megjegyezni, hogy bár a módszer lokális minimumot keres, ennél a konkrét  $E(\underline{v})$  függvényenél ez egybeesik a globális minimummal (Grimson [1]).

## 7.2.A template-ek megtervezése

A neurális hálózattal való megvalósítás a digitális számítógéppel való (pl. gradiens módszeren alapuló) minimumkereséshez képest számos előnnyel bír. Ezek közül a legfontosabb a nagy sebesség. Ez a ténylegesen megépített (nem csak szimulált) hálózatoknál jelentős részben a párhuzamosságból adódik. Hiszen  $m \times m$ -es ráccsal ábrázolt felületeknél  $m \times m$  elemi processzorsejt dolgozik párhuzamosan. A nagy sebesség másik oka az analóg számítási elv. Vagyis a digitális számítógépeknél szokásos címzések, adatmozgatások elmaradnak.

Tehát most a celluláris neurális hálózatra jellemző template-ek megtervezése következik. Olyan template-et választunk, hogy a hálózat az általunk elvárt differenciálegyenlet-rendszert oldja meg. A rácselemek magasságértékei most a sejtek belső állapotaiban vannak.

Először is vizsgáljuk meg a celluláris neurális hálózat matematikai modelljét (Roska [10], 87.o.). Egy sejt viselkedését a következő egyenlet írja le:

$$C \frac{dv_{xij}(t)}{dt} = \frac{-1}{R_x} v_{xij}(t) + \sum_{(k,l) \in N_r(i,j)} A(i,j,k,l) v_{ykl}(t) + \sum_{(k,l) \in N_r(i,j)} B(i,j,k,l) v_{ukl}(t) + I$$

$$v_{yij} = f(v_{xij})$$
(7.16)

$C$ ,  $R$  és  $I$  a sejthez tartozó kapacitás, ellenállás, és additív áramérték. A  $v_{xij}$ ,  $v_{yij}$  és  $v_{uij}$  az  $(i,j)$  pozícióban levő sejt belső állapota, kimenete, és a sejthez tartozó bemenet. Az  $A(i,j,k,l)$  és  $B(i,j,k,l)$  a kimenetek és a bemenetek csatolására vonatkozó súlytényezők. A rácselrendezés egy sejtje csak egy  $r$  sugarú környezetével van közvetlen kapcsolatban.

Ha a  $C=1$ ,  $R=1$  gyakran használt egyszerűsítésekkel élünk:

$$\frac{dv_{xij}(t)}{dt} = -v_{xij}(t) + \sum_{(k,l) \in N_r(i,j)} A(i,j,k,l) v_{ykl}(t) + \sum_{(k,l) \in N_r(i,j)} B(i,j,k,l) v_{ukl}(t) + I$$
(7.17)

Megjegyzendő, hogy itt az állapotváltozó változási gyorsaságát adja meg az egyenlet jobb oldala.

A gradiens módszerrel  $\underline{v}$  vektor két iteráció közti változását a következőképpen kapjuk:

$$\Delta \underline{v} = -\beta \text{grad } E(\underline{v})$$
(7.18)

A gradiens módszer lényege az, hogy a  $\underline{v}$  vektort mindig a negatív gradiens irányába változtatjuk, hiszen adott pontból kiindulva ebben az irányban csökken az  $E(\underline{v})$  függvény értéke ugyanakkora kis  $\Delta \underline{v}$  változásra a leginkább.

A minimalizálandó  $E(\underline{v})$  függvény, és ebből adódóan a gradiens alakja a következő:

$$E(\underline{v}) = \frac{1}{2} \sum_{(i,j)} \sum_{(k,l) \in N_r(i,j)} C(i,j,k,l) v_{(i,j)} v_{(k,l)}, \quad r = 2$$

$$\frac{\partial E}{\partial v_{(i,j)}} = \sum_{(k,l) \in N_r(i,j)} C(i,j,k,l) v_{(k,l)}, \quad r = 2 \quad (7.19)$$

$$C(i,j,k,l) = \frac{\partial^2 E}{\partial v_{(i,j)} \partial v_{(k,l)}}$$

Azonban CNN esetén nem az állapotváltozók két iteráció közti megváltozását, hanem a változási gyorsaságot tudjuk megadni. Így áll elő a következő alak:

$$\frac{d\underline{v}}{dt} = -\beta \text{grad} E(\underline{v}) \quad (7.20)$$

Megjegyzem, hogy a  $\beta$  itt is egy konstans, bár dimenziója más. A hasonló szerep miatt jelöltem ugyan azzal a betűvel. A celluláris neurális hálózatnál a magasságértékek a sejtek belső állapotaiban vannak tárolva. Így ezek után a  $v_{(i,j)}$  magasságérték helyét  $v_{xij}$  veszi át.

Egy rácselemre:

$$\frac{dv_{xij}}{dt} = -\beta \frac{\partial E}{\partial v_{xij}} \quad (7.21)$$

Mivel a gradiensről már megmutattuk, hogy egy-egy koordinátája a  $v_{xij}$  értékek lineáris kombinációja:

$$\frac{dv_{xij}}{dt} = -\beta \sum_{(k,l) \in N_r(i,j)} C(i,j,k,l) v_{xkl} \quad (7.22)$$

$$C(i,j,k,l) = \frac{\partial^2 E}{\partial v_{xij} \partial v_{xkl}}$$

ahol a  $C(i,j,k,l)$  együtthatókat ismerjük.

Az (7.4) és (7.22) egyenletet összehasonlítva megkapjuk, hogy az A, B, I értékek és a sejt kimenetét adó nemlineáris függvény miként alakulnak.

1.  $\underline{B} = 0$ , vagyis a  $v_{uij}$  bemenetek nem számítanak.
2.  $I = 0$ .
3.  $v_{yij} = v_{xij}$ , tehát  $f(x) = x$ .
4. Az ismert pontokhoz tartozó sejtek  $v_{xij}$  belső állapotaiba a felület adott pontbeli magasságát töltjük.
5. Az ismeretlen pontokhoz tartozó sejtekbe nullát töltünk.
6. A ismert pontokhoz tartozó sejtek belső állapotának változását letiltjuk.
7. A kialakított felületet a sejtek  $v_{xij}$  belső állapotaiban kapjuk, vagy az ezzel egyenlő  $v_{yij}$  kimenetekben.
8. Az energiaképletben  $r = 2$ , tehát  $\underline{A}$   $5 \times 5$ -ös.
9. Az  $\underline{A}$  mátrix elemei a következőképp alakulnak :

$$A(i, j, k, l) = \begin{cases} -\beta C(i, j, k, l), & \text{ha } (i, j) \neq (k, l) \\ -\beta C(i, j, k, l) + 1, & \text{ha } (i, j) = (k, l) \end{cases} \quad (7.23)$$

A +1  $(i, j) = (k, l)$  esetén azért szükséges, mivel a CNN (7.4) egyenletében a jobb oldalon már eleve szerepel egy  $-v_{xij}$ -s tag.

Sajnos többfajta template adódik, a template értékek nem lesznek helyfüggetlenek.  $\beta=1$  esetén ezek a következők:

$$a.) \begin{vmatrix} 0 & 0 & -2 & 0 & 0 \\ 0 & -4 & 16 & -4 & 0 \\ -2 & 16 & -39 & 16 & -2 \\ 0 & -4 & 16 & -4 & 0 \\ 0 & 0 & -2 & 0 & 0 \end{vmatrix} \quad b.) \begin{vmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -4 & 12 & -4 & 0 \\ -2 & 16 & -37 & 16 & -2 \\ 0 & -4 & 16 & -4 & 0 \\ 0 & 0 & -2 & 0 & 0 \end{vmatrix} \quad (7.24)$$

$$c.) \begin{vmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -2 & 12 & -21 & 12 & -2 \\ 0 & -4 & 12 & -4 & 0 \\ 0 & 0 & 2 & 0 & 0 \end{vmatrix} \quad d.) \begin{vmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 8 & -19 & 12 & -2 \\ 0 & -4 & 12 & -4 & 0 \\ 0 & 0 & -2 & 0 & 0 \end{vmatrix} \quad (7.25)$$

$$e.) \begin{vmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -4 & 12 & -4 & 0 \\ 0 & 12 & -35 & 16 & -2 \\ 0 & -4 & 16 & -4 & 0 \\ 0 & 0 & 2 & 0 & 0 \end{vmatrix} \quad f.) \begin{vmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -7 & 8 & -2 \\ 0 & 0 & 8 & -4 & 0 \\ 0 & 0 & -2 & 0 & 0 \end{vmatrix} \quad (7.26)$$

Elhelyezkedésük az  $m \times m$ -es rácson:

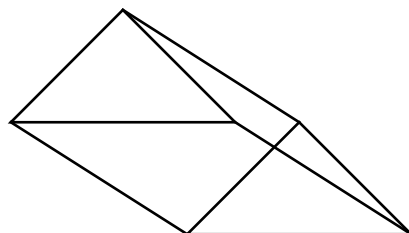
i / j	0	1	2	3	4	.	m - 3	m - 2	m - 1
0	f	d	c	c	c	.	c	dy	fy
1	dz	e	b	b	b	.	b	ey	dzy
2	cz	bz	a	a	a	.	a	bzy	czy
3	cz	bz	a	a	a	.	a	bzy	czy
4	cz	bz	a	a	a	.	a	bzy	czy
.	.	.	.	.	.	.	.	.	.
m - 3	cz	bz	a	a	a	.	a	bzy	czy
m - 2	dx	ex	bx	bx	bx	.	bx	exy	dxxy
m - 1	fx	dx	cx	cx	cx	.	cx	dxy	fxxy

### 7.2. Táblázat

A táblázatban az első betű az (7.24), (7.25), (7.26) alatt felsorolt template-eket jelöli, az "x", "y", "z" pedig tükrözést jelent az x, y tengelyre, illetve az x és y tengellyel 45 fokos szöget bezáró egyenesre. Ebből látható, hogy összesen 25 féle template szükséges.

Lássuk a hálózat működését egy egyszerű példán. A felületet  $m \times m$ -es rácscsal jellemeztük, ahol  $v_{ij}$  a felület alapsíktól mért magassága az (i,j) pontban. A celluláris neurális hálózatunk is  $m \times m$  sejtéből áll. Most az (i,j) pozíciójú sejt belső állapota jelenti a felület magasságát az alapsík (i,j) pontjától. Így reprezentáljuk a felületet a hálózattal.

A következő példán egy oldalán fekvő háromszög alapú hasáb látható.



7.3. Ábra

A két felső lap által alkotott felület így jelenik meg egy 5\*5-ös struktúrában:

$i \setminus j$	0	1	2	3	4	
0	1	2	3	2	1	
1	1	2	3	2	1	
2	1	2	3	2	1	
3	1	2	3	2	1	
4	1	2	3	2	1	(7.27)

Felület illesztésére a következőképpen használhatjuk a tervezett hálózatot. Ha az alapsík (i,j) pontjában ismerjük a felület magasságát, az (i,j) pozíciójú sejt belső állapotába ezt a magasságértéket töltjük, és a sejt állapotváltozását letiltjuk. Ha nem ismert a felület magassága ebben a pontban, a sejtbe nullát töltünk. A következő ábra egy olyan esetben mutatja be a celluláris neurális hálózat sejtjeibe töltendő értékeket, mikor csak három pontot ismerünk.

$i \setminus j$	0	1	2	3	4	
0	0	0	3	0	0	
1	0	0	0	0	0	
2	1	0	0	0	0	
3	0	0	0	0	0	
4	0	0	3	0	0	(7.28)

Ezután elindítjuk a hálózatot. Kis idő múlva megkapjuk a pontokra illesztett felületet. Három ismert pont esetén ez várhatóan síklap. És tényleg egy síklapot írunk le a sejtek belső állapotai az iteráció végén.

$i \setminus j$	0	1	2	3	4	
0	1	2	3	4	5	
1	1	2	3	4	5	
2	1	2	3	4	5	
3	1	2	3	4	5	
4	1	2	3	4	5	(7.29)

Megjegyzendő, hogy amennyiben a celluláris neurális áramkört szimuláló program csak egyféle template alkalmazását teszi lehetővé, egyszerűsítésekkel mégis létrehozható ez a fajta hálózat. Ilyenkor a legegyszerűbb lehetőség az "a"-val jelölt template-et használni, és a szimulátor azon opcióját beállítani, amikor a szabadon maradt bemenetekre nullát ad. Ez avval jár, hogy a kialakuló felület szélei két sorban a nulla szinten vannak rögzítve.

Valamivel több lehetőséget nyújt az a megoldás, amikor a szélső két sorban letiltjuk a sejtek állapotánál változását és a nekünk megfelelő értékre állítjuk ezt a belső állapotot. Vagyis a rács szélső két sorában minden pontot ismertnek adunk meg.

Ha a szimulátor rendelkezik egy speciális opcióval, az ún. kettőzéssel, egy harmadik módszer is kínálkozik. A szimulátor ilyenkor a szélső oszlopban/sorban levő sejteket a széleken megismétli. Vagyis a sejtekből álló négyzetrács körül létrejön egy virtuális sejtsor. Itt egy sejt kimenete megegyezik a velük szomszédos, valódi sejt kimenetével. Értelemszerűen ez a módszer két virtuális sejtsorral is használható. Így a szélen nem kell nekünk meghatározni a felület pontjait. A módszer hátránya, hogy a felület a szélen (a szélre merőleges vízszintes által meghatározott irányban) vízszintes érintőre fog törekedni, akkor is, ha ezt az ismert pontok nem indokolják.

### 7.3.Template tervezés más módszerekkel

Az előbbi gondolatmenet mellett másképpen is meg lehet kapni a template-eket. Erre ismertetek röviden két módszert. Ez azért is tanulságos, mert új összefüggésekre világít rá.

#### 7.3.1.Template tervezés a biharmonikus egyenlettel

Mint azt a matematikai modell Grimson féle tárgyalásánál megállapítottuk, a legsimább felületet jellemző  $v(x,y)$  függvény kielégíti a biharmonikus egyenletet:

$$\nabla^4 v = \frac{\partial^4 v}{\partial x^4} + 2 \frac{\partial^4 v}{\partial x^2 \partial y^2} + \frac{\partial^4 v}{\partial y^4} = 0 \quad (7.30)$$

Ha a határgörbe  $x$ - $y$  síkra való vetülete négyzet alakú, Grimson [3]szerint a következő peremfeltételek érvényesek:

$$\frac{\partial^2 f}{\partial y^2} = 0, \quad \frac{\partial^3 f}{\partial x^2 \partial y} = 0 \quad (7.31)$$

az  $x$  tengellyel párhuzamos határolóoldal mentén, és

$$\frac{\partial^2 f}{\partial x^2} = 0, \quad \frac{\partial^3 f}{\partial y^2 \partial x} = 0 \quad (7.32)$$

az y tengellyel párhuzamos határolóoldal mentén.

Először a bal oldalon szereplő operátort kell diszkretizálni, az  $E(v)$  függvény diszkrét alakjához hasonlóan. Ez ugyanúgy megkapható, mint a rácselemekhez tartozó magasságértékek függvénye. Legyen a  $v_{(i,j)}$  felületen értelmezett, diszkretizált operátor jele a nagy D betű. Ezek után a neurális hálózattal a

$$\frac{dv_{(i,j)}}{dt} = \pm Dv_{(i,j)} \quad (7.33)$$

egyenletet kell megvalósítani. Ez azért oldja meg a biharmonikus egyenletet, mert stabil állapotban a derivált nulla. A jobb oldal előjelét stabilitási megfontolásokból kell levezetni. Az, hogy van stabil végállapot, bizonyításra szorul. (Nyilván a gradiens módszerrel működő hálózat is ezt számítja ki, az pedig stabil. )

Ez volt tehát a biharmonikus egyenleten alapuló levezetés vázlata. Ez azonban kapcsolatban áll a gradiens módszerrel számítottakkal. A gradiens módszerrel ezt az egyenletet valósítottuk meg:

$$\frac{dv_{xij}}{dt} = -\beta \frac{\partial E}{\partial v_{xij}} \quad (7.34)$$

Ebből látszik, hogy a két jobboldal csak egy skalár szorzóban térhet el egymástól.

### 7.3.2.Template tervezés a neurális hálózat energiaképletével

A template-ek egyszerűen megkaphatók, ha felhasználjuk a celluláris neurális hálózat energiafüggvényét. Ez a Hoppfield-féle hálózat energiafüggvényével (Roska [10], 142-145.o.) megegyezik, csupán a kapcsolatok lokálisak.

$$E = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n T_{ij} V_i V_j + \sum_{j=1}^n \int_0^{V_j} \frac{g^{-1}(V)}{R_j} dV - \sum_{i=1}^n V_i I_i \quad (7.35)$$

Itt  $n$  a neuronsejtek száma,  $V_i$  az  $i$ . sejt kimenete,  $g(x)$  a belső állapotból a kimenetet adó nemlineáris függvény,  $I_i$  az  $i$ . sejtbe befolyó áram,  $R_i$  az  $i$ .



sejthez tartozó ellenállásérték. A  $T$  együtthatók azt határozzák meg, mekkora súllyal van az  $i$ . sejt kimenete a  $j$ . sejt bemenetére visszacsatolva.

Tekintsük azt az esetet, amikor a sejtekbe befolyó áram 0,  $g(x)=x$  (vagyis a kimenet egyenlő a sejt belső állapotával), az ellenállás pedig minden sejtre  $R=1$ . Ezenkívül figyelembe véve, hogy az  $m \times m$  négyzetrácson elhelyezkedő sejteket kettős indexszel láttuk el, és hogy a celluláris neurális hálózatnál a  $T$  tényezők megfelelője  $A(i,j,k,l)$ :

$$E = -\frac{1}{2} \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} \sum_{k=0}^{m-1} \sum_{l=0}^{m-1} A(i, j, k, l) v_{xij} v_{xkl} + \frac{1}{2} \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} v_{xij}^2 \quad (7.36)$$

Felelevenítve a gradiens módszer során kiszámolt energiafüggvényt:

$$E(\underline{v}) = \frac{1}{2} \sum_{(i,j)} \sum_{(k,l) \in N(i,j)} C(i, j, k, l) v_{(i,j)} v_{(k,l)}, \quad r = 2 \quad (7.37)$$

Ahhoz, hogy a két energiafüggvény minimuma ugyanott legyen, megengedett, hogy egy konstans szorzóban különbözzenek. Az egyszerűség érdekében ettől tekintsünk el, és válasszuk a két függvényt megegyezőnek. A neurális hálózatot meghatározó  $A$  értékeit ekkor így kapjuk:

$$A(i, j, k, l) = \begin{cases} -C(i, j, k, l), & \text{ha } (i, j) \neq (k, l) \\ -C(i, j, k, l) + 1, & \text{ha } (i, j) = (k, l) \end{cases} \quad (7.38)$$

Ebből az  $A$  mátrix ugyanúgy adódik, mint azt az előző pontban kiszámoltuk. Ez a hálózat egyéb jellemzőire is igaz. A  $B$  mátrix 0, a  $I$  áramérték szintén, a belső állapotból a kimenetet az  $f(x)=x$  függvény állítja elő.

#### 7.4. Az áramkör működése néhány példa kapcsán

Az előbbi jellemzőkkel rendelkező celluláris neurális áramkör szimulálására megfelelő számítógépprogramot készítettem, amely az előrelépő Euler formulával oldotta meg a differenciálegyenletet. A közelítés  $C=1$ ,  $R=1$  esetén 0,015-es időlépésnél még a helyes megoldáshoz konvergált, 0,02-es értéknél már divergens volt. A felületek ábrázolásához  $40 \times 40$ -es rácsot használtam.

A következő példák bemutatják, hogyan függ a konvergencia gyorsasága az ismert pontok sűrűségétől és elhelyezkedésétől, milyen hibát okoz egy rosszul ismert pont, és hogyan viselkedik a rendszer a felület hirtelen meredekségváltozásánál.

Az első példa azt mutatja, milyen felületet illeszt a hálózat három nem egy egyenesbe eső pontra. Három ponton át mindig fektethető egy síklap. Egyenlete

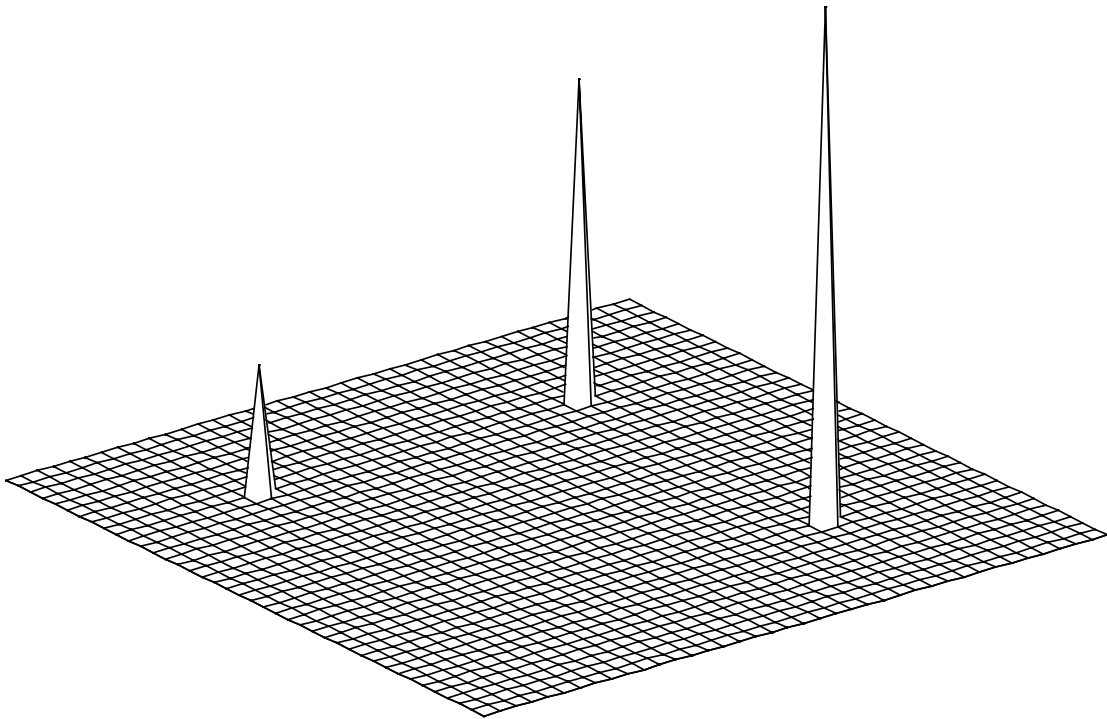
$$z = ax + by + c. \quad (7.39)$$

Így

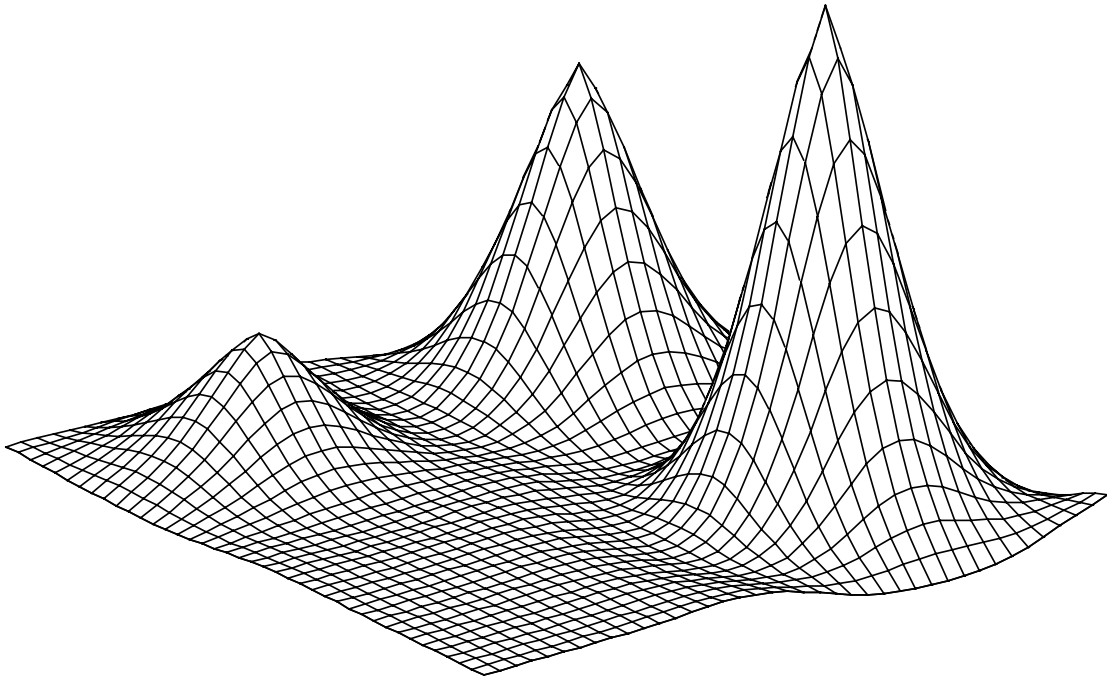
$$\frac{\partial^2 v}{\partial^2 x} = 0, \quad \frac{\partial^2 v}{\partial^2 y} = 0, \quad \frac{\partial^2 v}{\partial x \partial y} = 0 \quad (7.40)$$

Ebből az adódik, hogy a síklap esetén az energiafüggvény 0. Tehát három pont esetén a síklap a minimumot adó felület.

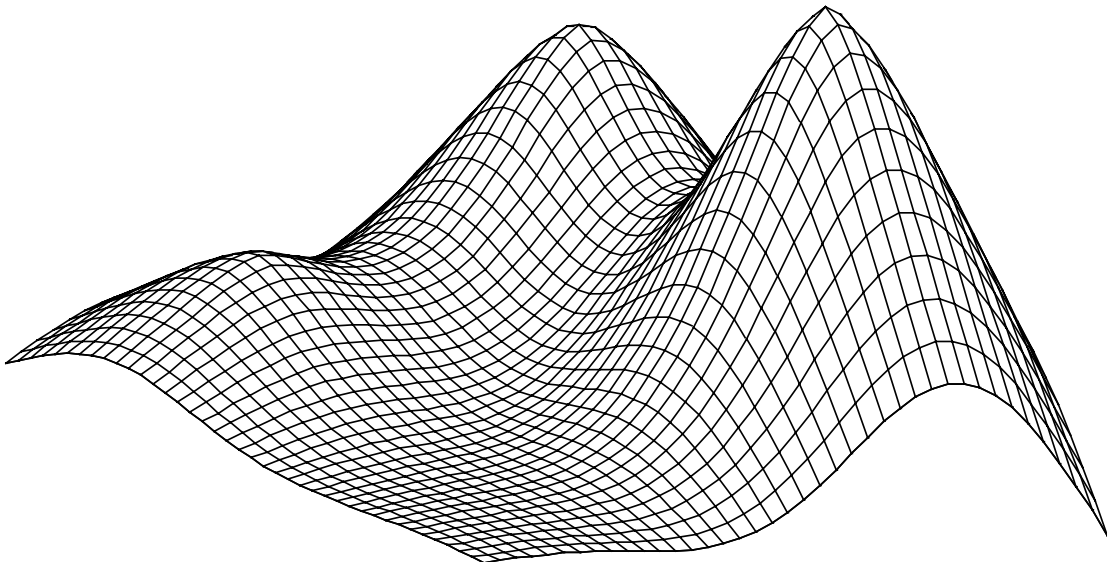
A 7.1.a. ábra a kiinduló helyzetet mutatja, a 7.1.b, c, d, e ábra pedig az 15, 150, 750, 17550 időegység utáni állapotot.



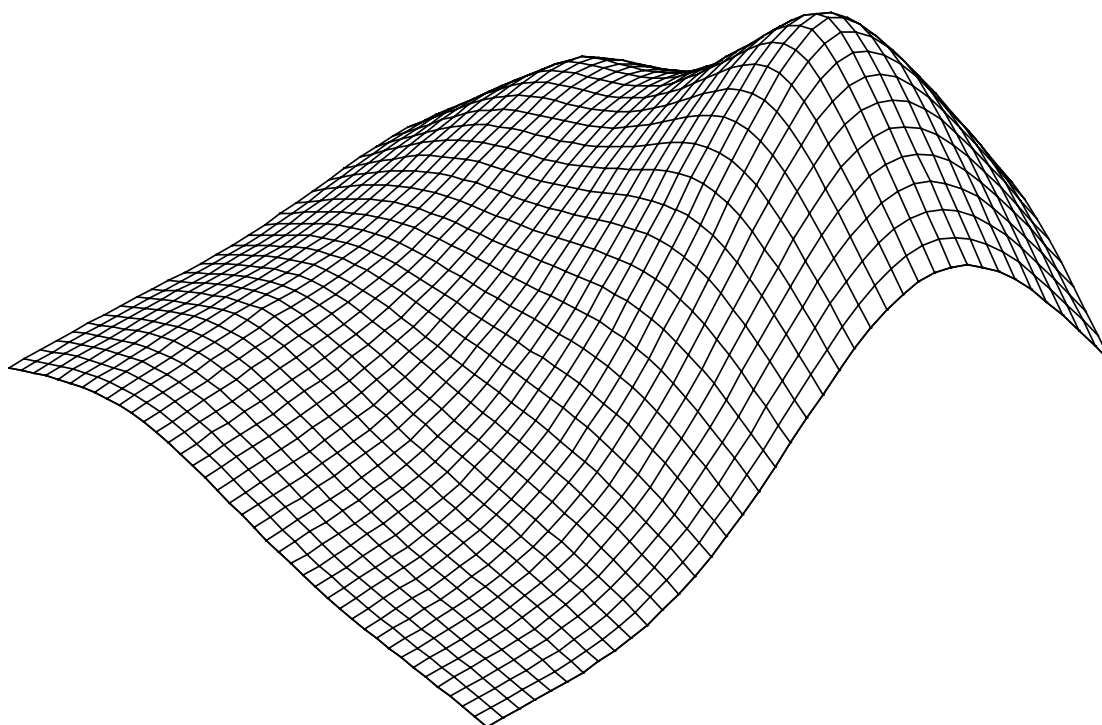
7.1.a. Ábra



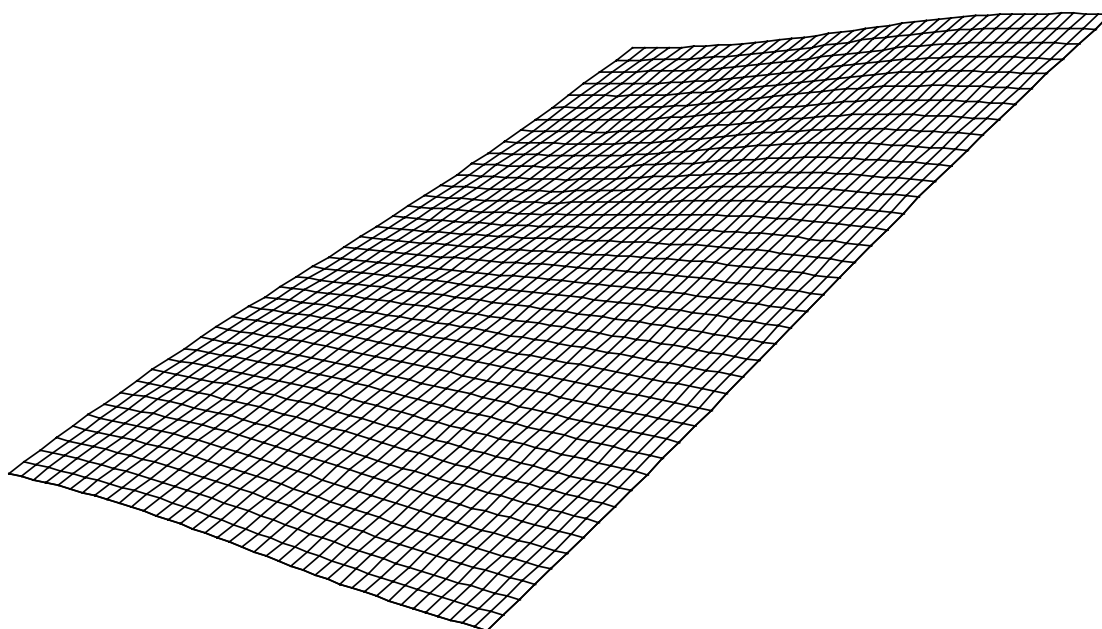
7.1.b. Ábra



7.1.c. Ábra

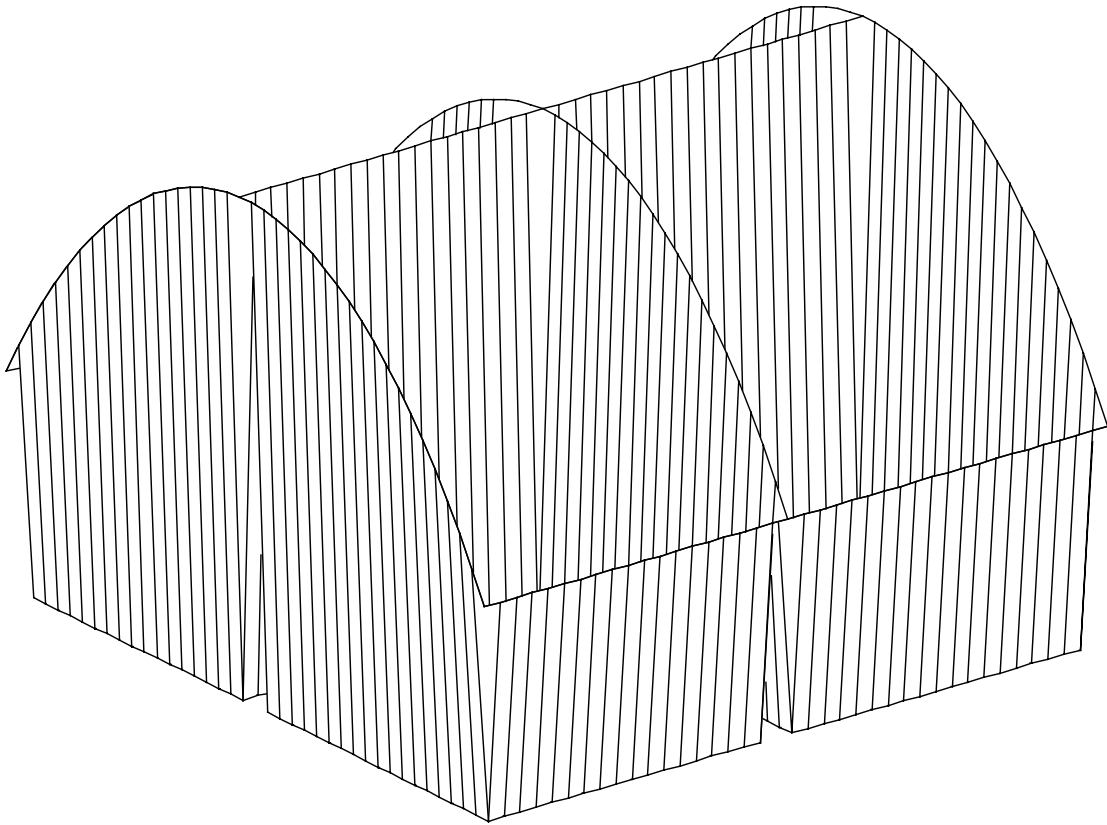


7.1.d.Ábra

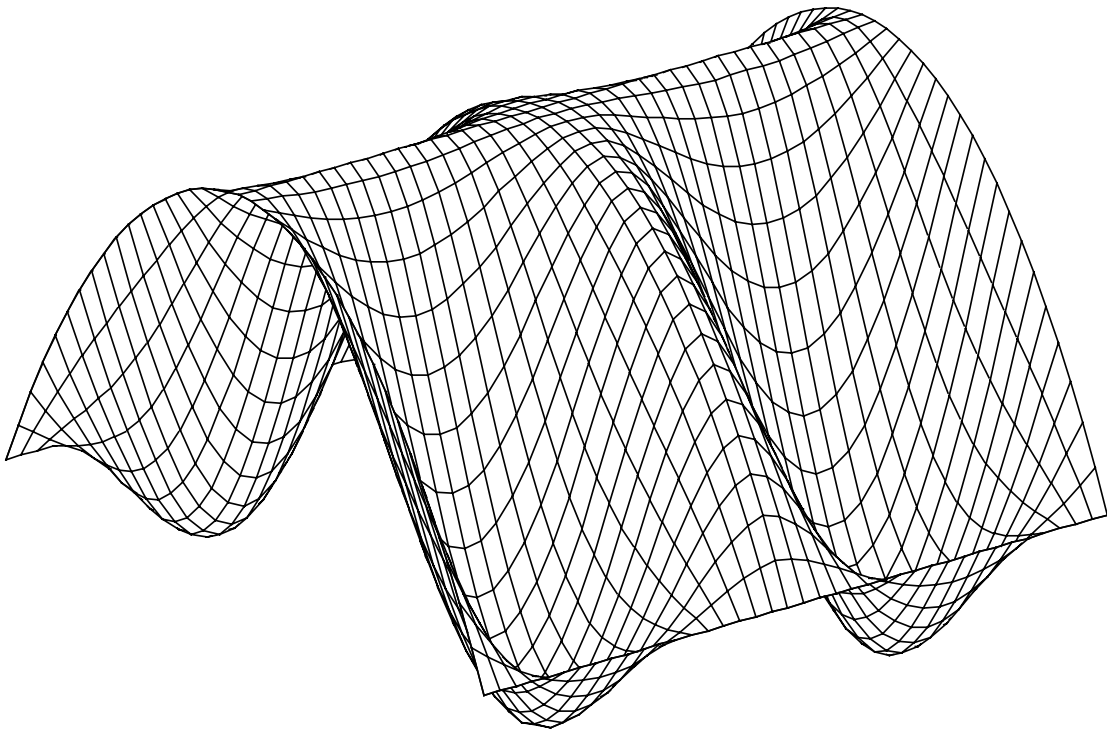


7.1.e.Ábra

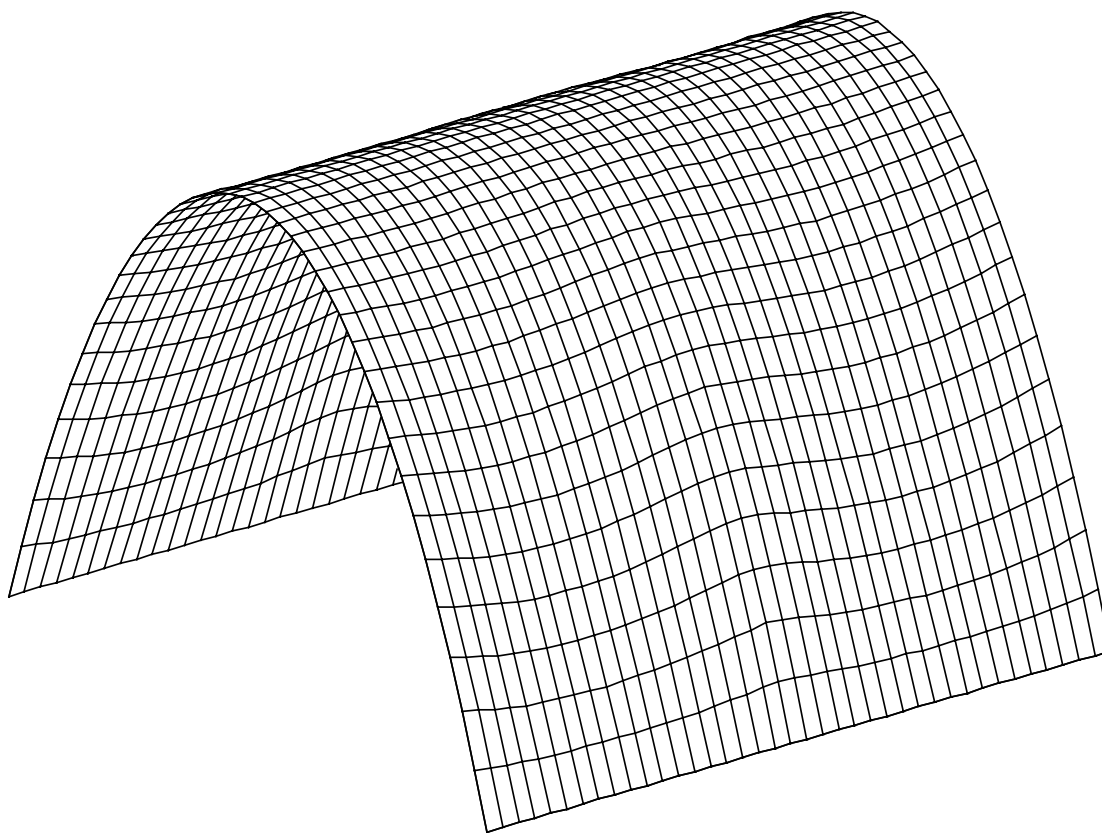
A következő példa egy hiányosan definiált hengerpalástrészlet rekonstrukcióját mutatja. A 7.2.a ábrán az ismert pontok, a 7.2.b és a 7.2.c ábrán a CNN struktúra által 100 és 850 időegység után kialakított felület látható.



7.2.a.Ábra

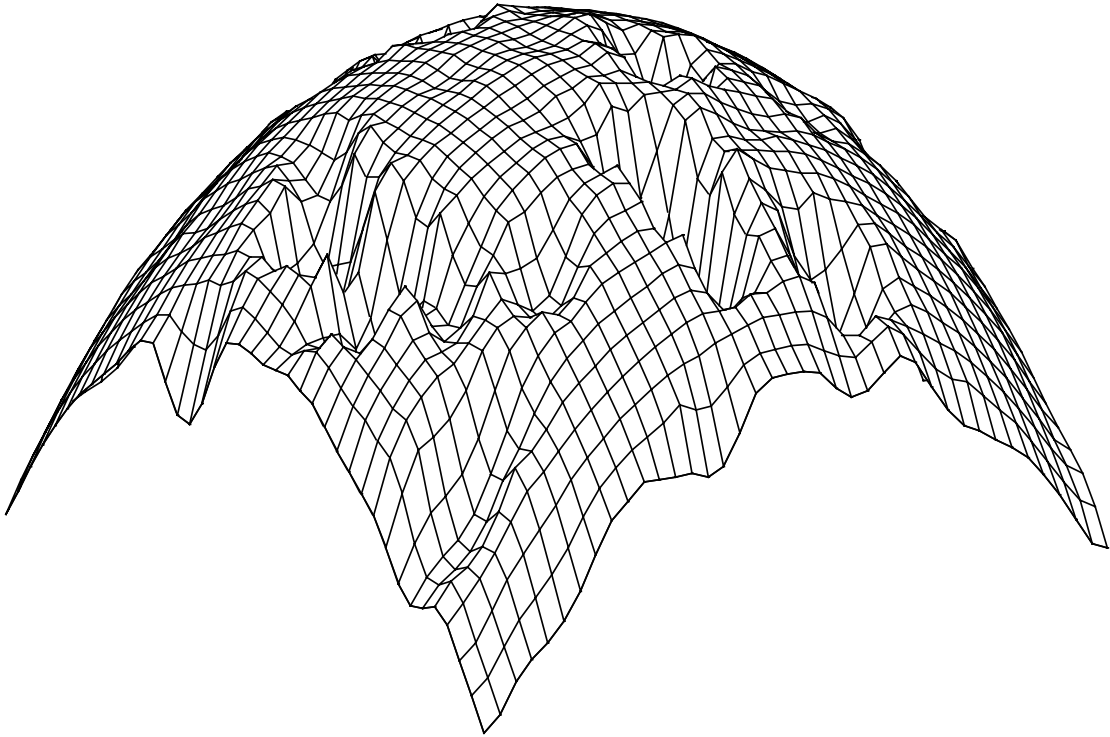


7.2.b.Ábra

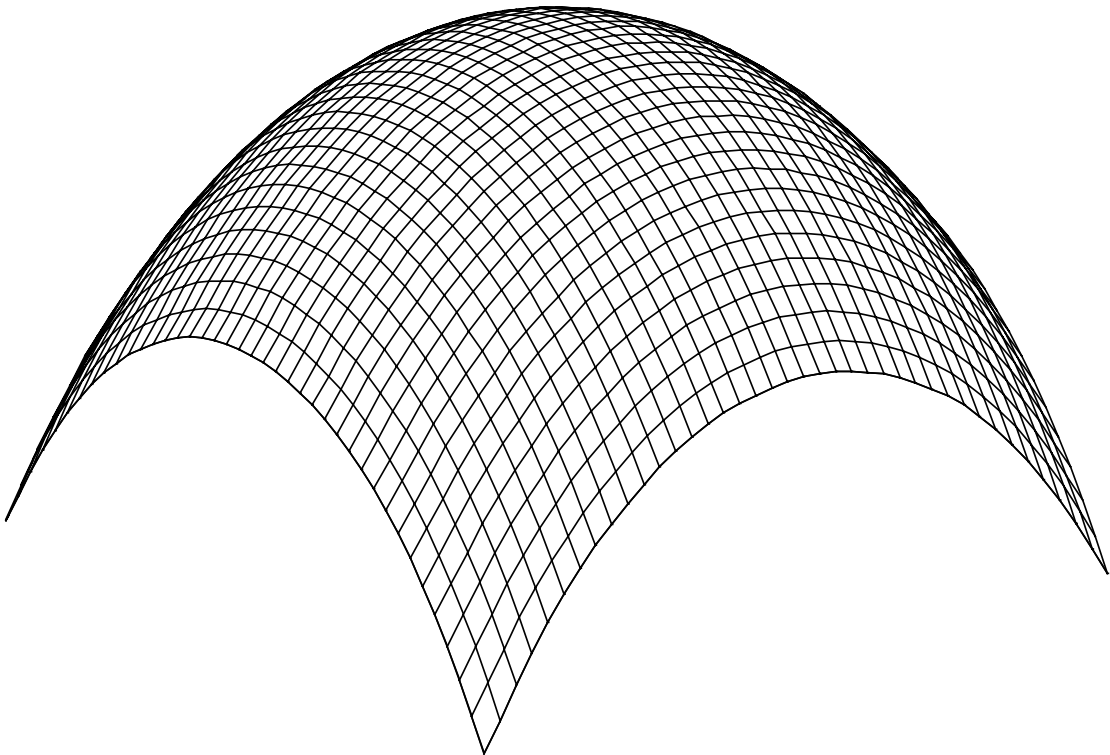


7.2.c. Ábra

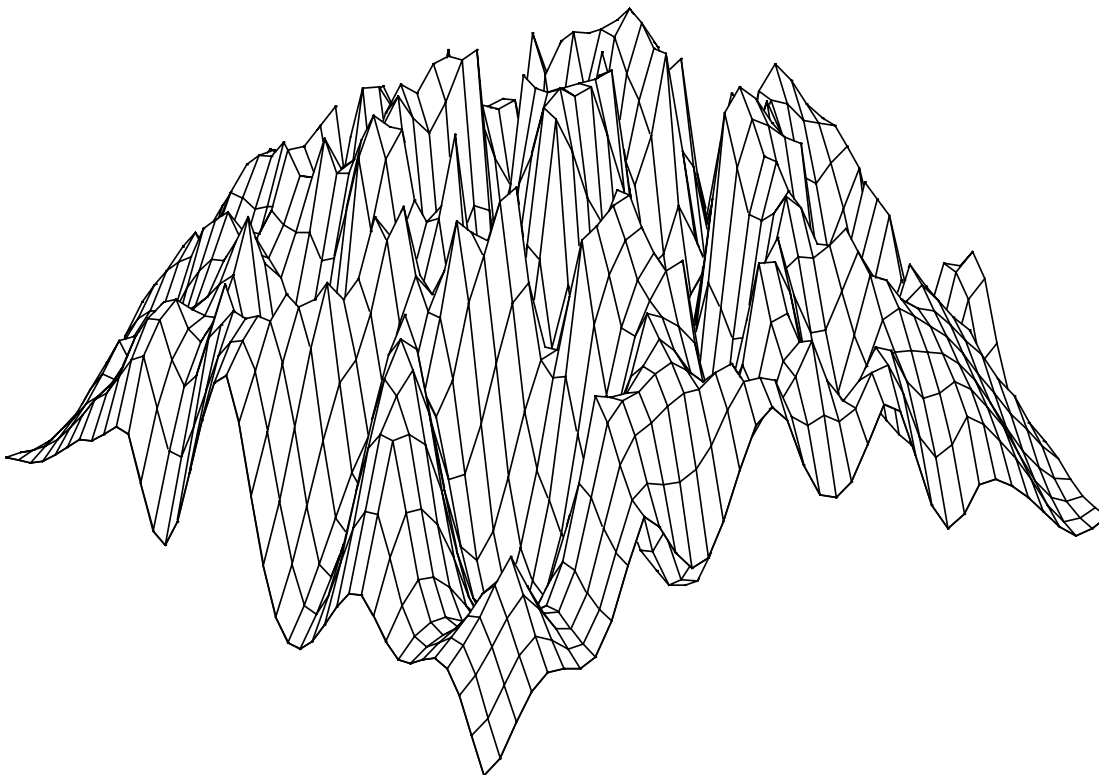
A harmadik példa egy hiányosan definiált gömbfelület esetén mutatja az áramkör működését. A 7.3.a. ábrán látható helyzethez csupán 0,75 időegységre volt szükség. A 15 időegység után előálló végeredményt a 7.3.b. ábra mutatja. Az ismert pontok sűrűsége 30%. A feladatot kipróbáltam úgy is, hogy az ismert pontok sűrűsége 10% és 3% volt. A 7.3.c. ábra a 10%-os esetben mutatja a 0,75 időegység után előálló eredményt. A 7.3.d. ábrán és a 7.3.e. ábrán pedig az ismert pontok 3%-os sűrűsége esetén 0,75 és 75 időegység után előálló helyzet látható. Érdekes az összehasonlítás a 7.3.a, c és d ábra között. Látható, hogy a 30, 10 és 3%-os esetben 0,75 időegység után hova jut el a rendszer. Az utóbbi esetben az erre az időre kialakuló kép nem túl biztató, ennek ellenére még így is sikerült a felület rekonstrukciója 150 időegység után.



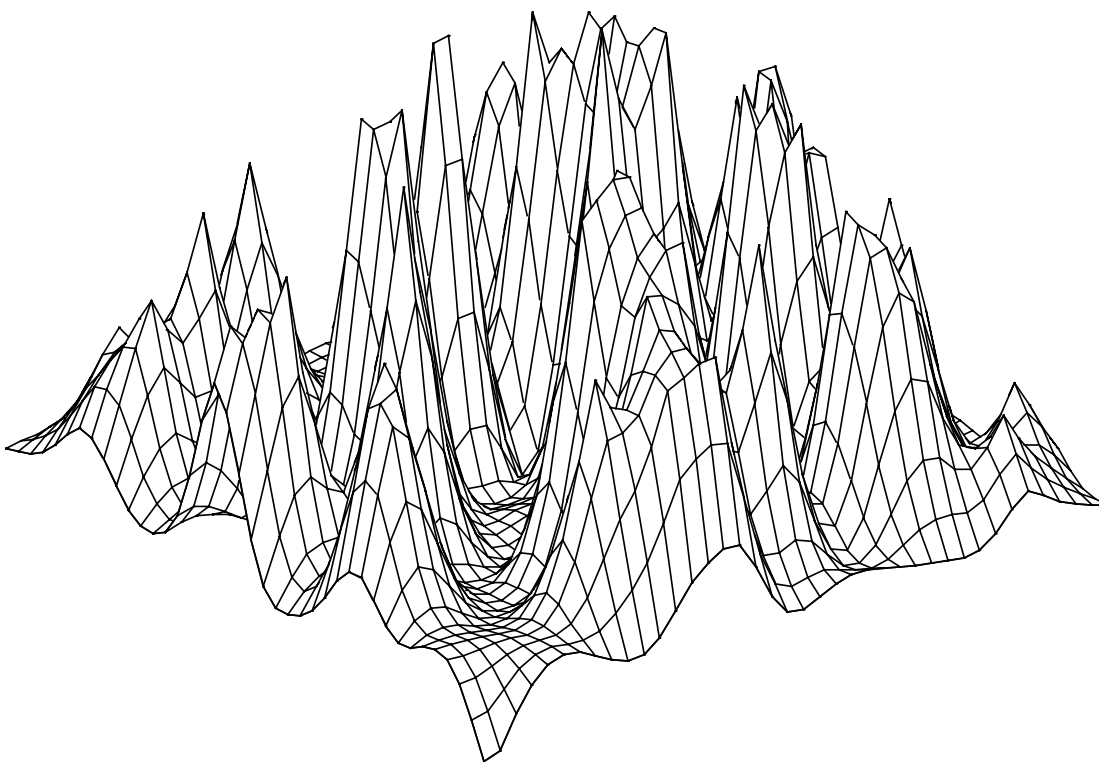
7.3.a.Ábra



7.3.b.Ábra

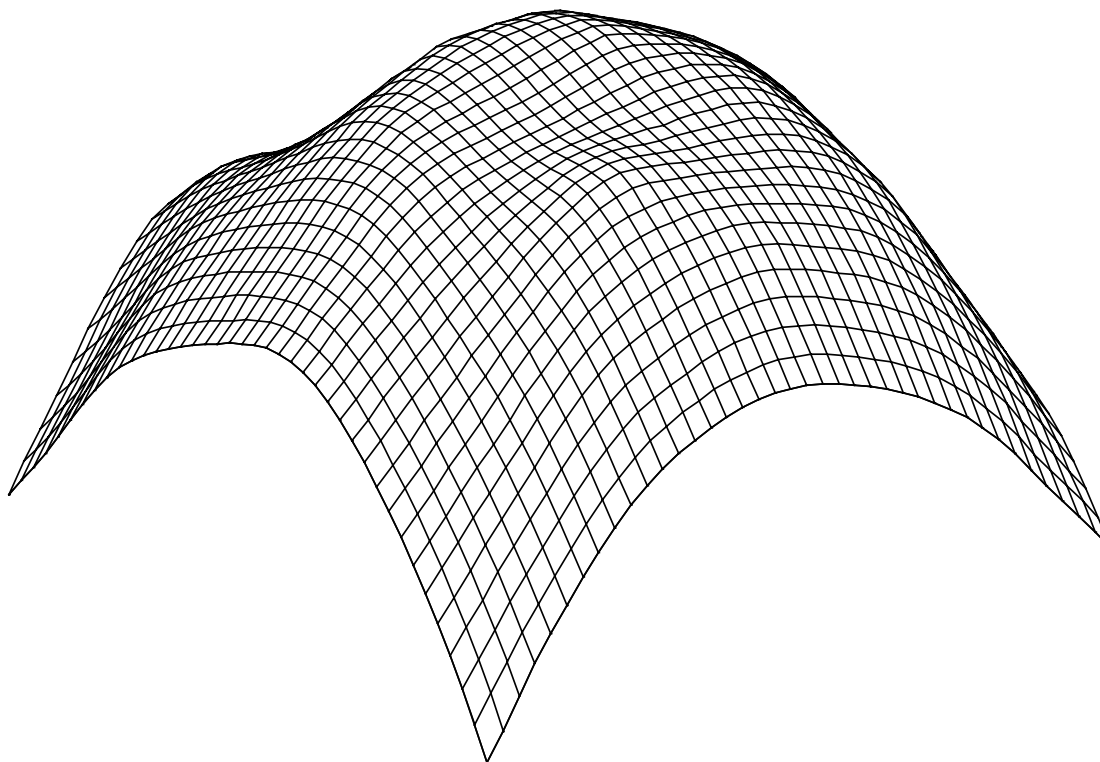


7.3.c. Ábra



7.3.d. Ábra

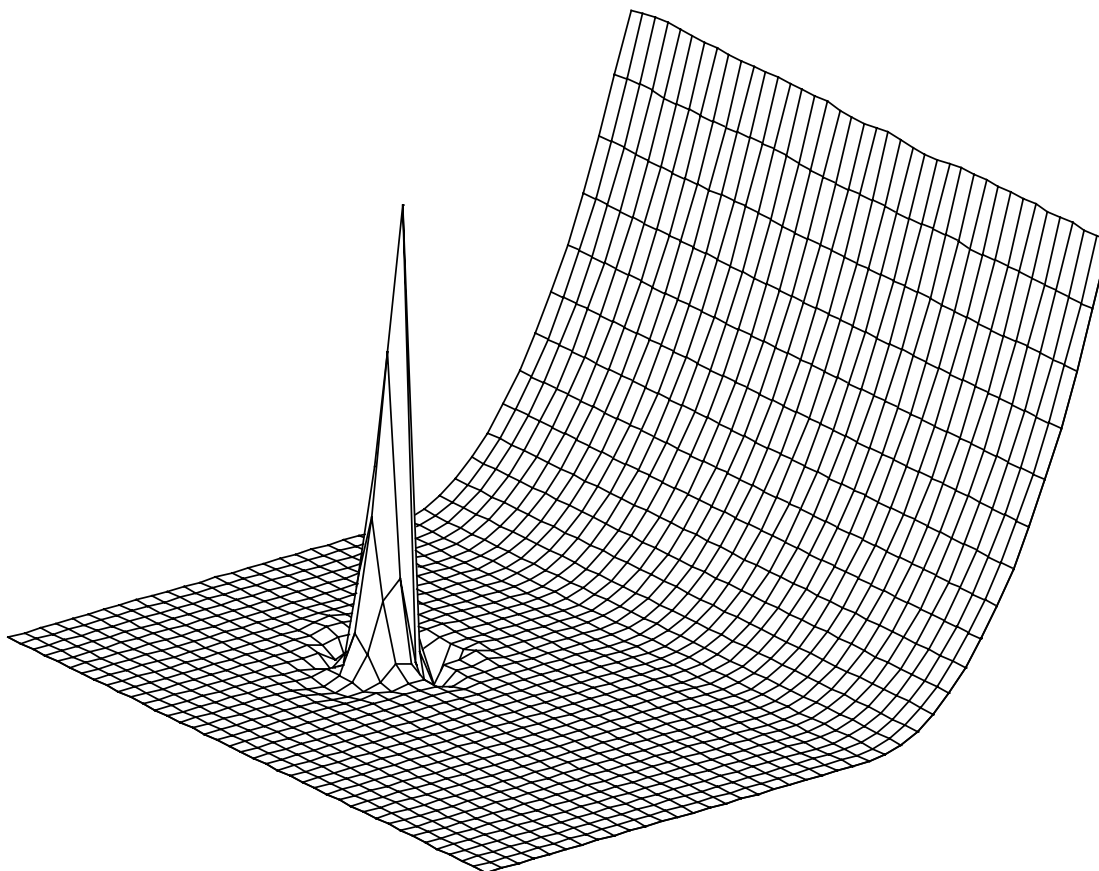




7.3.e.Ábra

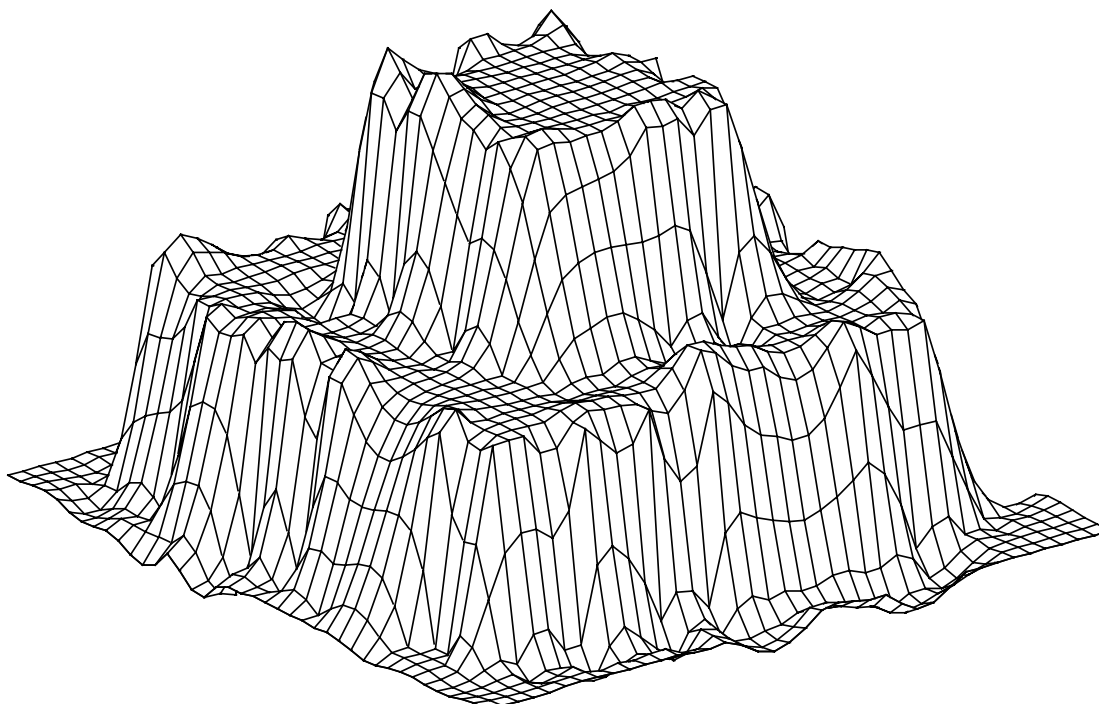
A két példa közti lényeges eltérés, az hogy a hengerpalástrészletnél az ismert pontok sűrűsége az ábra egyes részein különböző. Egyes egyenesek mentén minden pontot ismerünk, nagy területek azonban ismeretlenek. A félgömb esetén az ismert pontok sűrűsége az egész ábrán 30%. Ennek a konvergencia gyorsaságára van kihatása, mivel a nagy ismeretlen területek csak hosszú idő alatt alakulnak ki megfelelőképpen a módszer lokalitásának köszönhetően. Másként fogalmazva, a lokalitás következtében a hiba (a végállapot és a jelenlegi állapot közti különbség) kis térfrekvenciás komponense igen nehezen tűnik el.

A negyedik példa a 7.4.ábrán azt mutatja hogyan viselkedik a módszer akkor, ha egy felület rosszul megadott pontot is tartalmaz. Ez valós tárgy háromdimenziós képének rekonstrukciójakor gyakran előfordul, mivel nincs 100%-os biztonságú eljárás a két vetületi képből az ismert pontok kiszűrésére. Látható, hogy az algoritmus a megadott pontokon mindenképpen átvezeti a felületet, ezért egy rosszul megadott pont szerencsétlen esetben az egész ábrát jelentősen torzíthatja. Az ismert pontok sűrűsége egyébként 30% volt, az áramkör működési ideje 15 egység.



7.4. Ábra

A következő példa a 7.5. ábrán látható esküvői torta ("wedding cake") alakzat. Ez alapesetnek számít a randompont sztereogramoknál, hiszen ott gyakoriak az olyan alakzatok, amelyek az alapsíkkal párhuzamos síkokból jönnek létre. Az ismert pontok sűrűsége 30%, a kép létrejöttéhez szükséges idő 16,5 egység. A síklapok belső részei viszonylag hamar kialakulnak, a probléma a síklapok határán jelentkezik. Itt bizonyos "túllövés" tapasztalható. A módszer hátránya, hogy hirtelen meredekségváltozást nehezen tud áthidalni. Másrészt a rácsrepresentáció amúgy sem alkalmas függőleges felületek ábrázolására.

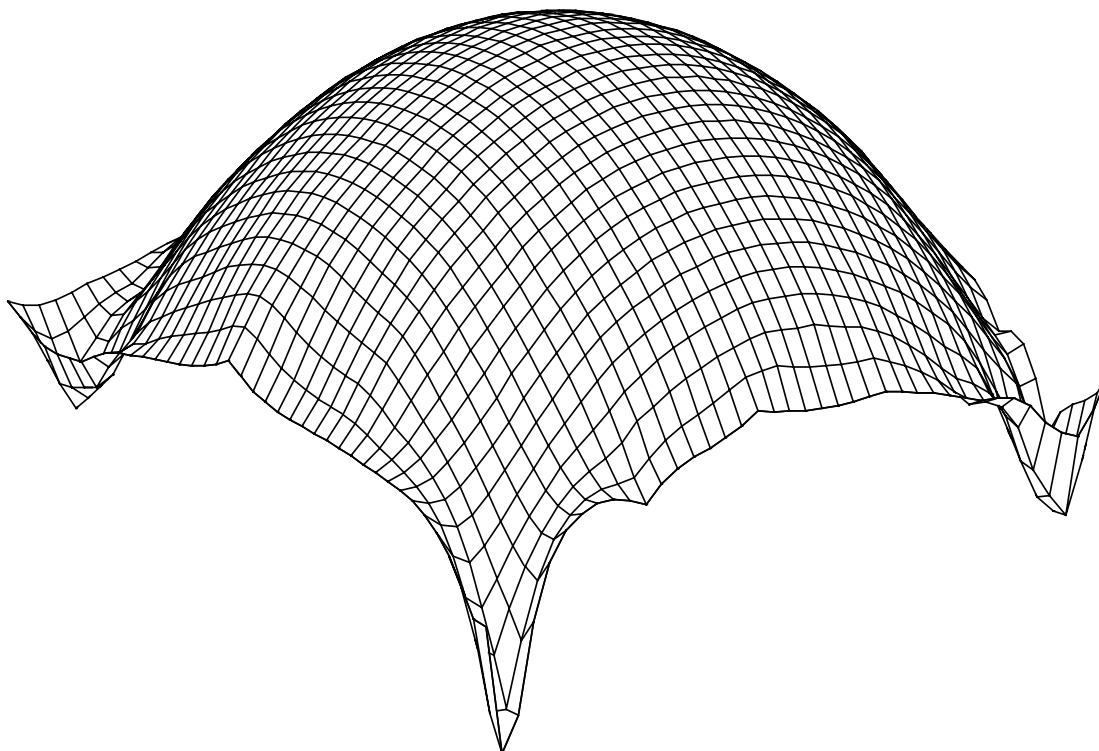


7.5.Ábra

Az általam tervezett neurális hálózatot kipróbáltam a CNNM V5.3-mas szimulátorral is. (Bővebben lásd [10].) Ennek van egy egész számokkal és egy lebegőpontos számokkal megvalósított verziója. Az előbbi nem elég pontos, mivel a törtrész ábrázolásához 9 bitet használ, az az  $1/512$  a legkisebb ábrázolható érték, a változások pedig a kis időlépés (0,015) miatt kicsik. Így a lebegőpontos változatot alkalmaztam.

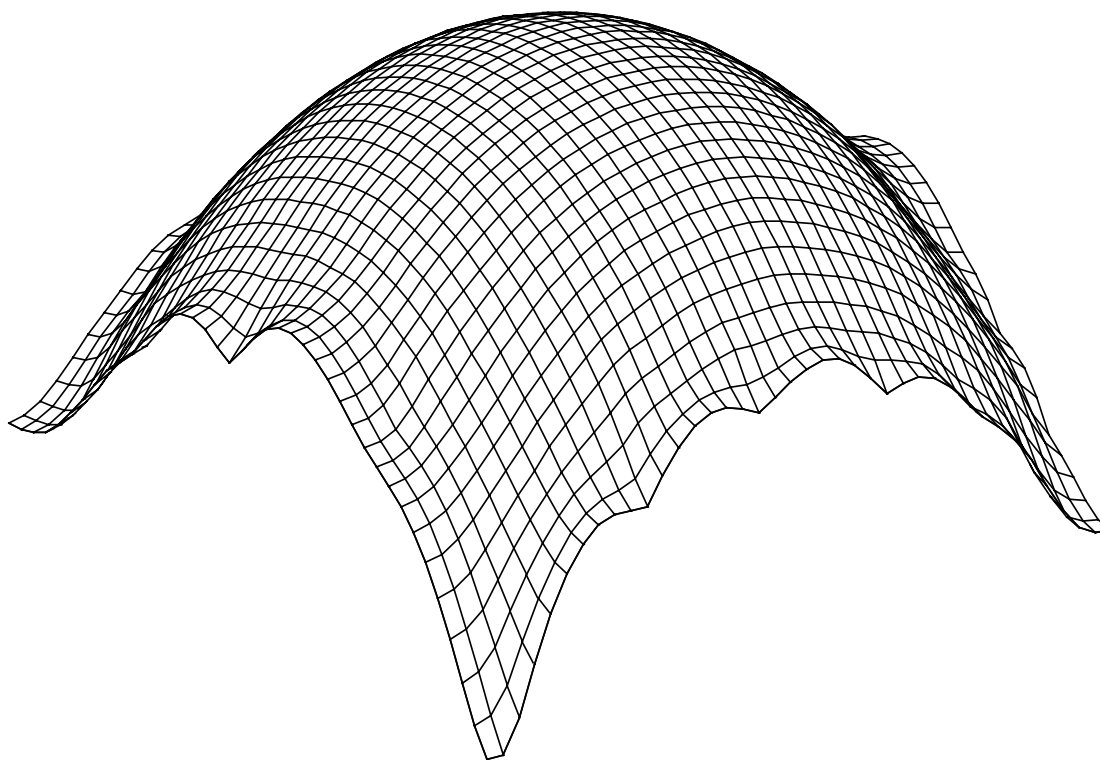
A template tervezésről szóló fejezet végén már írtam a szimulátoros megvalósításról, illetve arról, hogyan lehet egy template alkalmazásával mégis kielégítő eredményt kapni. Ezt illusztrálom a 10%-os ismert pontsűrűséggel rendelkező gömbfelület 45 időlépés utáni rekonstrukcióján.

A 7.6.a.ábrán a szélen levő sejtek szabadon maradt lábaira 0-át adattam. A gömbfelület magasságértékei  $-0,9$  és  $+0,9$  között mozogtak, mivel a szimulátor olyan, hogy ha a sejtek belső állapota a  $[-1,+1]$  intervallumban van, a sejtek kimenete megegyezik a bemenettel. Tehát ez a sejtekben lévő nemlinearitás lineáris szakasza. Azzal, hogy a szélekre nullát adtunk, a hálózat a felület szélét a nulla szintre igyekszik kötni, ez látható is az ábrán.



7.6.a. Ábra

A 7.6.b. ábrán a már említett kettőzést próbáltam ki. Ekkor a széleken a szélső sort/oszlopot a szimulátor megismétli. A hálózat ilyenkor a felület szélén (a szélre merőleges vízszintes által meghatározott irányban) vízszintes érintőre törekszik. Ez látható is az ábrán, hiszen a felület széle indokolatlanul felhajlik.



7.6.b.Ábra

## **8.Az approximációt megvalósító neurális hálózat**

Az előző interpolációs eljárásnál az illesztett felület feltétlenül átment minden ismert ponton. A gyakorlatban ennek sok hátránya van. Randompont sztereogrammoknál, és általában akkor, ha két vetületi képből kívájuk a háromdimenziós alakzatot rekonstruálni, az ismertnek feltételezett pontok jelentős része a megvalósított algoritmusok esetén tévedés eredménye. Tehát az ismert háromdimenziós pontokat meghatározó eljárások a gyakorlatban nem nyújtanak teljes biztonságot. Az első interpolációs eljárás pedig egy rosszul meghatározott ponton akkor is átvezetné a felületet, ha ez az ábra jelentős torzulását eredményezné.

A most ismertetésre kerülő módszer kiküszöböli ezt a hibát. Az ismert pontokra való illeszkedést nem úgy biztosítjuk, hogy a felületnek feltétlenül át kell mennie rajtuk, hanem úgy, hogy a minimalizálandó függvényben a simaságot kifejező első tag mellett megjelenik az ismert pontokra való illeszkedést jellemző második tag. Azt a modellt használjuk, amelyben a vékony lapot rugókkal rögzítettük, nem pedig mereven. Így a kiszámított potenciális energiaképlet mindkét tagját alkalmazzuk.

### **8.1.A gradiens módszer**

A függvény minimumának megkeresésére Grimson [1] a konjugált gradiens módszert alkalmazta. Ez csupán annyiban különbözik a gradiens módszertől, hogy nem a negatív gradiens irányában lép, hanem a lépés irányának kiszámításában az előző iteráció irányát is felhasználja. Így gyorsabb a konvergencia. A celluláris neurális hálózathoz a gradiens módszert tartottam implementálásra alkalmasnak.

Így a minimalizálandó energiafüggvény:

$$E(v(x, y)) = \iint \left( \frac{\partial^2 v}{\partial x^2} \right)^2 + 2 \left( \frac{\partial^2 v}{\partial x \partial y} \right)^2 + \left( \frac{\partial^2 v}{\partial y^2} \right)^2 dx dy + \sum_{(x_i, y_i) \in I} \alpha [v(x_i, y_i) - c_{(x_i, y_i)}]^2 \quad (8.1)$$

Ennek diszkretizált alakja:

$$E(\underline{v}) = \sum_{i=1}^{m-2} \sum_{j=0}^{m-1} (v_{(i-1, j)} - 2v_{(i, j)} + v_{(i+1, j)})^2 + \sum_{i=0}^{m-1} \sum_{j=1}^{m-2} (v_{(i, j-1)} - 2v_{(i, j)} + v_{(i, j+1)})^2 + 2 \sum_{i=0}^{m-2} \sum_{j=0}^{m-2} (v_{(i, j)} - v_{(i+1, j)} - v_{(i, j+1)} + v_{(i+1, j+1)})^2 + \alpha \sum_{(i, j) \in I} (v_{(i, j)} - c_{(i, j)})^2 \quad (8.2)$$

Az  $\alpha$  konstans itt azt határozza meg, hogy az optimális felület keresésénél milyen súllyal vegyük figyelembe az ismert pontokra illeszkedést a simasághoz képest. Minél nagyobb, a felület annál inkább illeszkedni fog az ismert pontokra.

Most a gradiens meghatározása a feladat. Ehhez új alakra hozzuk az energiafüggvényt a zárójelek felbontásával.

$$E(\underline{v}) = \frac{1}{2} \sum_{(i, j)} \sum_{(k, l) \in N(i, j)} C(i, j, k, l) v_{(i, j)} v_{(k, l)} + \alpha \sum_{(i, j) \in I} (v_{(i, j)} - c_{(i, j)})^2, \quad r = 2$$

$$C(i, j, k, l) = C(k, l, i, j) \quad (8.3)$$

Ebből adódóan a  $v_{(i,j)}$  szerinti derivált, vagyis a gradiens vektor egy koordinátája, ha az  $(i,j)$  rácspontban a  $c_{(i,j)}$  magasság ismert:

$$\frac{\partial E}{\partial v_{(i,j)}} = \sum_{(k,l) \in N_r(i,j)} C(i,j,k,l) v_{(k,l)} - 2\alpha c_{(i,j)} + 2\alpha v_{(i,j)}, \quad r=2 \quad (8.4)$$

Ha az  $(i,j)$  pontban nem ismert a magasság, a jobb oldali összegnek csupán első tagja szerepel.  $C(i,j,k,l)$  ugyanúgy alakul, mint interpolációnál. Így az ott kapott táblázatok használhatóak.

## 8.2.A template-ek megtervezése

A neurális hálózatot jellemző differenciálegyenlet  $C=1$ ,  $R=1$  esetén:

$$\frac{dv_{xij}(t)}{dt} = -v_{xij}(t) + \sum_{(k,l) \in N_r(i,j)} A(i,j,k,l) v_{ykl}(t) + \sum_{(k,l) \in N_r(i,j)} B(i,j,k,l) v_{ukl}(t) + I_{ij} \quad (8.5)$$

Látható, hogy az  $I$  áramérték helyfüggő. Erre a celluláris neurális hálózatoknál szokatlan tulajdonságra szükség lesz később.

A gradiens módszer neurális hálózatra alkalmazása pedig a következő egyenletet adja:

$$\frac{dv_{xij}}{dt} = -\beta \frac{\partial E}{\partial v_{xij}} \quad (8.6)$$

Ide a deriváltra előbb kapott képletet helyettesítve:

$$\frac{\partial v_{xij}}{\partial t} = -\beta \left( \sum_{(k,l) \in N_r(i,j)} C(i,j,k,l) v_{xkl} - 2\alpha c_{(i,j)} + 2\alpha v_{xij} \right), \quad r=2 \quad (8.7)$$

A hálózat jellemzői (8.5) és (8.7) összehasonlításából  $\beta=1$  esetén a következőképp adódnak:

1. Az abszolút pontos illesztési módszertől csupán azoknál a sejteknél különbözik, amelyek olyan rácsponthoz tartoznak, amelynél a felület magassága ismert.

A különbség az  $\underline{A}$  mátrix középső elemében és  $I_{ij}$  - ben van.

2. Ekkor az  $(i, j)$  rácsponthoz tartozó sejt  $\underline{A}$  mátrixának középső eleme:

$$a_{ij}^{(3,3)} = -C(i, j, i, j) - 2\alpha$$

3. Ugyanennél a sejtnél  $I_{ij}$  :

$$I_{ij} = 2\alpha c_{(i,j)}$$

(8.8)

A gyakorlatban ez azt jelenti, hogy ezekben a pontokban az  $A$  mátrix annyiban különbözik az interpolációnál kapott  $A$  mátrixtól, hogy a középső eleméből le kell vonjunk  $2\alpha$ -t, a sejthez tartozó áramérték pedig nem 0, hanem a fenti képlet szerint alakul. Ez az áramérték az ismert pont magasságától és  $\alpha$ -tól függ.

Megjegyzem, hogy ezt a módszer csak egyfajta template-et lehetővé tevő szimulátorral már megkötések mellett sem lehet használni. Sőt ha többféle template-et is megengedő programot alkalmazunk, az ismert pontoknak akkor is befolyása van a template értékekre és az áramértékekre. Vagyis egy-egy feladathoz külön template-összeállítást kell készíteni. Így egy külön a feladat szimulálására szolgáló program a helyzetet jelentősen egyszerűsíti.

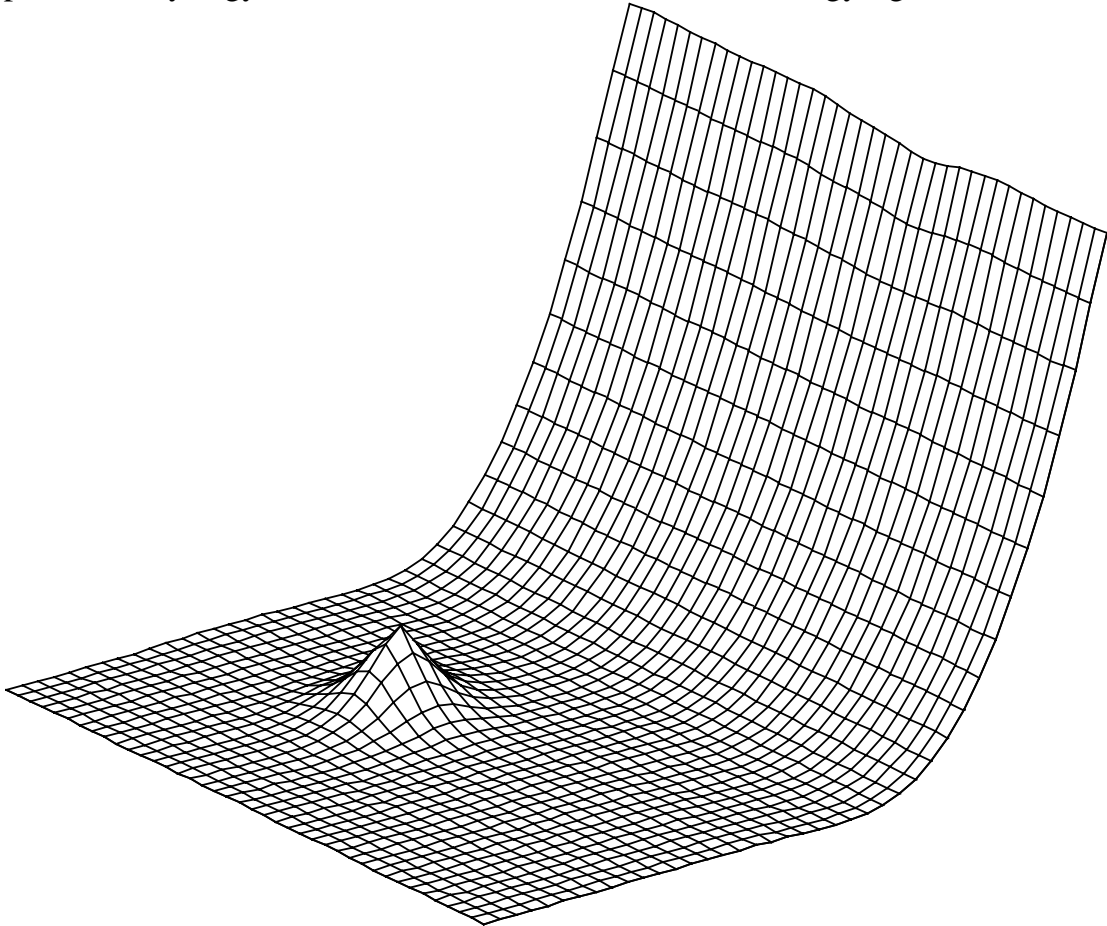
### 8.3. Az áramkör működése néhány példa kapcsán

A következő példák olyan eseteket mutatnak, amikor az áramkör által kialakított felület jelentősen különbözik az interpoláció esetén kapott eredménytől.

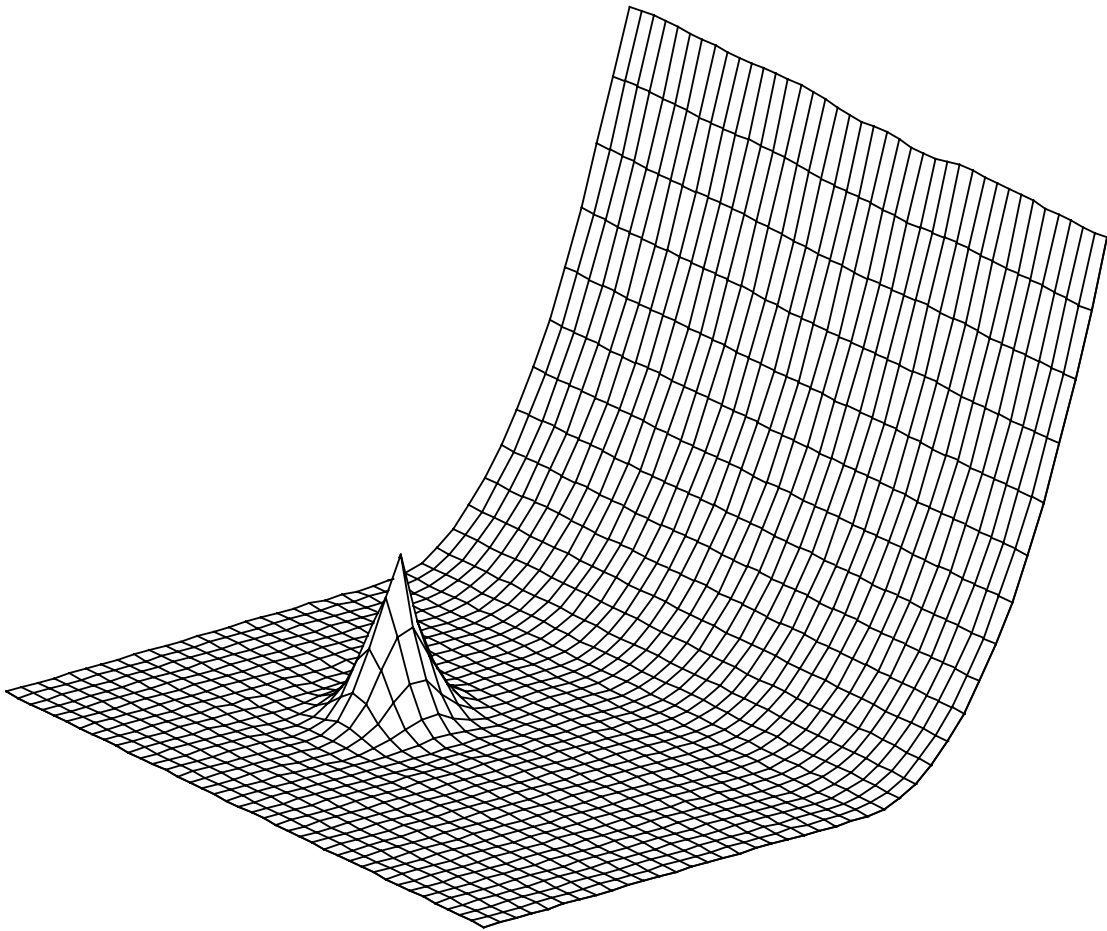
Az első példán olyan felület rekonstrukcióját mutatjuk be, amelyen egy pontot hibásan adtunk meg. Ezt az problémát már a másik illesztési módszernél is tárgyaltuk. Akkor ez az egy rosszul megadott pont az egész ábrát elrontotta. Most sokkal kisebb a torzulás. A legkisebb az 8.1.a. ábrán látható esetben,



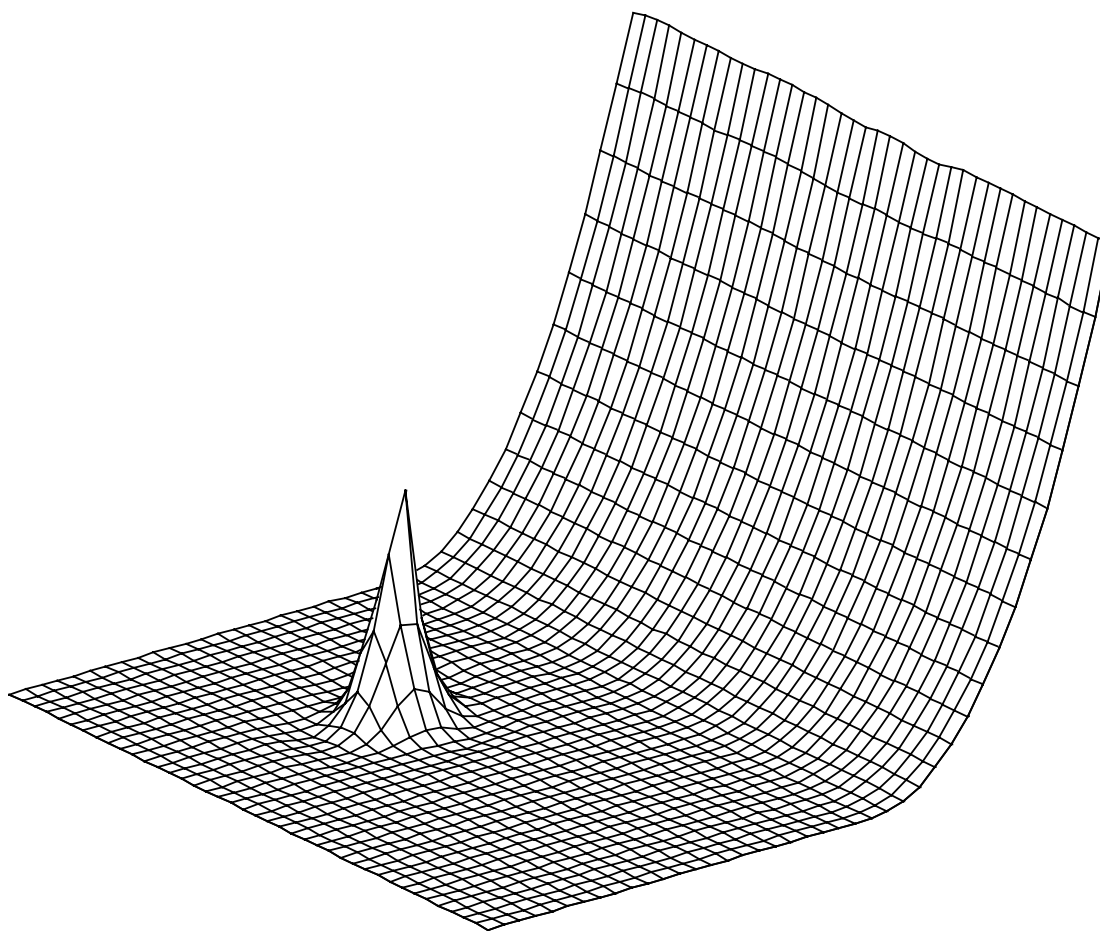
hiszen itt az illesztettség pontosságát jellemző  $\alpha$  konstans is a legkisebb, csupán 0,5. Az 8.1.b.ábrán a hibás pont már jobban zavar, itt  $\alpha=1,5$ . Az 8.1.c.ábrán látható esetben pedig ez a kellemetlen hatás még nagyobb, itt  $\alpha=5$ . Az ismert pontok aránya egyébként 30% volt, a működési idő 30 időegység.



8.1.a.Ábra



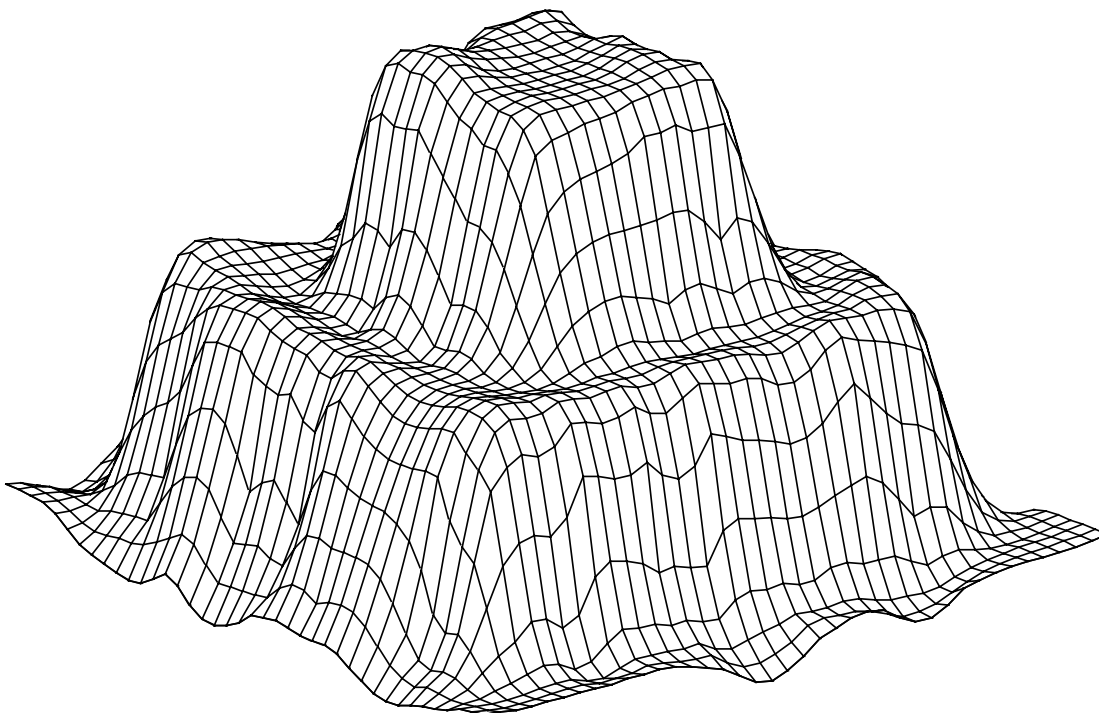
8.1.b.Ábra



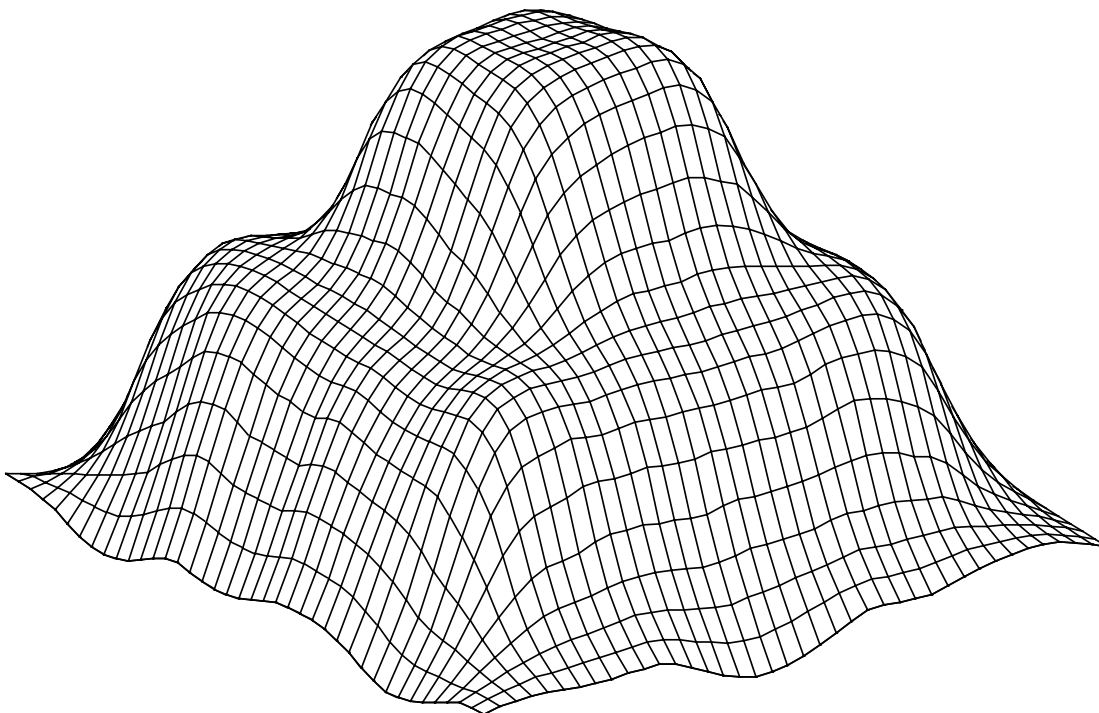
8.1.c.Ábra

Ebből látható, hogy  $\alpha$  minél nagyobb, a módszer annál inkább közelít az interpolációhoz. Hiszen az egyes felületeket értékelő  $E(\underline{y})$  függvényben a nagy  $\alpha$  miatt már az ismert pontoktól való kis eltérés is nagy additív tagot produkál. Azt a felületet keressük, amelyre az  $E(\underline{y})$  függvény értéke a legkisebb. A nagy additív tag miatt azonban a rosszul illeszkedő felületek kiesnek a minimumért való küzdelemből.

A másik példa a lakodalmi süteményt mutatja. A 8.2.a.ábránál  $\alpha=5$  volt. Az interpolációhoz képest a végeredmény valamivel kedvezőbb. A 8.2.b.ábra  $\alpha=0,5$  esetén mutatja a kialakult felületet. Az  $\alpha$  konstans, így az illeszkedés súlya az  $E$  függvényben túl kicsi volt. Így a mexikói piramisból majdnem egyiptomi piramis lett. Az ismert pontok sűrűsége 30%, a futási idő 45 időegység volt.



8.2.a.Ábra



8.2.b.Ábra

## **9.Összefoglalás**

Habár háromdimenziós világban élünk, az ember szemében kétdimenziós kép keletkezik. Mégis képesek vagyunk a tárgyakat felismerni, térbeli elrendezésüket megérteni. Vagyis az ember a két szemében keletkező kétdimenziós képből képes egy alakzat háromdimenziós képének rekonstrukciójára. Ez a sztereolátás.

Készítsünk fényképet két, párhuzamos optikai tengellyel rendelkező fényképezőgéppel, amelyek egymástól bizonyos távolságra helyezkednek el. Vajon hogyan tudnánk a képpáron szereplő tárgy térbeli alakját rekonstruálni számítógépes módszerrel? A feladatot analizálva azt találjuk, hogy az első elemi feladat minél több tárgypont távolságának meghatározása. Ehhez az kell, hogy a két képen azonosítani tudjuk az azonos tárgyponthoz tartozó két képpontot. Minél nagyobb az egyik képen levő képpont elmozdulása a másik képen levőhöz képest, annál közelebb van a hozzájuk tartozó tárgypont. Az elmozdulásból a tárgypont távolsága egyébként egy egyszerű képlettel számítható. (Ezt a képletet a bevezetőben meg is adom.)

Eddig csak egyes pontok térbeli helyét ismerjük, végeredményként azonban a lefényképezett tárgy felületét várjuk. Ezért a tárgy térbeli képének rekonstrukciójakor a második elemi feladat ezekre az ismert pontokra egy felületet illeszteni. Diplomamunkám ezzel az illesztési problémával foglalkozik. De mindezt nem számítógépes programmal kell megvalósítanom, hanem celluláris neurális hálózattal. (A celluláris neurális hálózatról bővebb ismeretek találhatók a függelékben.)

Az illesztés két alapvető módszerét vizsgáltam. Az egyszerűbb esetben a felületnek pontosan át kell haladnia az ismert pontokon. Ezt nevezzük *interpolációnak*. Végtelen olyan felület van, amely áthalad az ismert pontokon. Ezek közül kell egyet, mint optimálisat kiválasztani. Nyilván jó, ha ez a felület minél simább. Ennek a tulajdonságnak kellett matematikai megfogalmazását megtalálnom.

Az előző módszer nagy hátránya, hogy egy rosszul megadott pont az egész felületet eltorzíthatja, ugyanis a megadott pontokon a felületet feltétlenül átvezeti. Ezen segít az illesztés egy bonyolultabb esete, amikor az illesztett felületnek nem kell pontosan áthaladnia az ismert pontokon. Ezt *approximációnak* hívjuk. Elvárjuk, hogy a felület minél közelebb haladjon el az ismert pontokhoz, másrészt hogy minél simább legyen. Ez a két feltétel gyakran ellentmond egymásnak. Ezért a két elvárás súlyát a felület kiválasztásában meg kell adnunk.

Diplomamunkám első részében a matematikai modellválasztás problémáját tekintem át. Majd külön-külön az előbbi két módszer esetén tárgyalom az ún. gradiens módszer segítségével megvalósítandó algoritmust, és annak alkalmazását celluláris neurális hálózatra.

Interpoláció esetén olyan celluláris neurális hálózat adódott, amelynél a sejtek nem egyformák. Azonban megkötések árán használható azonos sejtekből álló hálózat is.

Approximáció esetén a hálózat bonyolultabb. Itt már az ismert pontok elhelyezkedése is befolyásolja a sejteket jellemző ún. template-ek értékét.

A tervezett áramkör működését konkrét példák kapcsán mutatom be. Ehhez megfelelő szimulátorprogramot készítettem.

## **10.Summary(összefoglalás angol nyelven)**

Although our world has three spatial dimensions, the projection of light rays onto the retina presents our visual system with an image of the world that is inherently two-dimensional. In spite of this we are able to recognize the objects in the three-dimensional world. So we can reconstruct the three-dimensional vision of the object from the two two-dimensional pictures. This is stereovision.

Let's make photos with two cameras. They have parallel optical axes and there is a distance between them. How could we reconstruct the three-dimensional shape of the object being on the photos using computer? If we analyse the problem, we can see, that the first thing to do is to find points on the two pictures, which belong to the same point on the object. The larger the difference between the position of the two points on the photos, the closer the point of the object, which belongs to them. The relationship between this difference and the distance of the point of the object can be easily computable by a simple formula, which is given in the introduction of this diploma work.

Now we know the spatial position of some points of the object, but we intend to get the surface of the object. So the second step of the reconstruction is to make surface that fits to this known points. The aim of my diploma work is to examine this fitting problem using cellular neural network (CNN).

Two basic methods are discussed in this work. In the surface *interpolation* problem we construct a surface that exactly fits the set of known points. There are innumerable surfaces that fit these points. We must choose the best one. There are several features that we could examine. The most important of these is smoothness. Therefore we choose the smoothest surface. So I had to

construct a mathematical function, which characterize the smoothness of the surface.

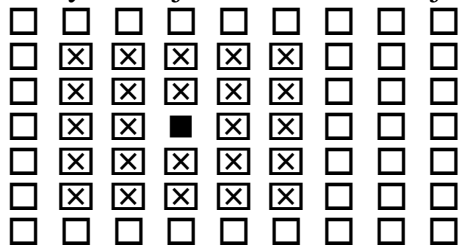
The real algorithms that collect the known points sometimes make mistakes. If there are some wrong points among our known points, then the surface given by the interpolation method fits exactly these points. So one bad point could deform the whole shape. The fitting problem can be relaxed somewhat into a surface *approximation* problem, by only requiring that the surface should approximately fit the known data and be smooth in some sense. The two conditions are sometimes inconsistent, so we use a weight-factor, that determine the weight of the two features in choosing the optimal surface.

In the first part of my diploma work I consider the problem of the mathematical model. The following part is written about the interpolation and approximation method. In both of these two sections I examine the mathematical background using the *gradient method*, the realization with cellular neural network, and the features of the networks with examples in simulator.

## 11.Függelék

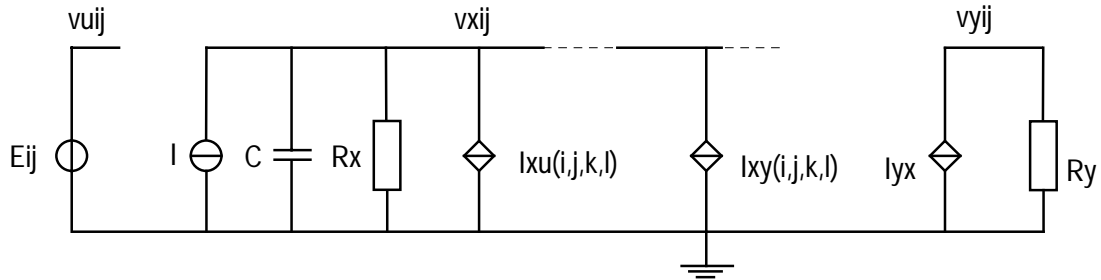
### 11.1.A celluláris neurális hálózat(CNN)

A celluláris neurális hálózat a kétdimenziós (háromdimenziós) neurális áramkörök közé tartozik. (Bővebben lásd a [10] irodalmat.) Egyrétegű esetben a neuronsejtek egy négyzetrács szögpontjaiban helyezkednek el. Mindegyik sejtre csupán adott sugarú környezete van közvetlen befolyással. A következő ábra egy ilyen elrendezést mutat. A fekete négyzettel jelölt sejtre csak az X-szel jelölt sejtek hatnak közvetlenül. Utóbbiak jelen esetben az  $r=2$  sugarú környezetét jelentik a fekete sejtnek.



11.1.Ábra

Egy sejt kapcsolási rajza a következő:



11.2.Ábra

A pozíciót az  $m \times n$ -es rácson két egész szám határozza meg (Pl.  $(i,j)$ ).

♦ A  $v_{xij}, v_{yij}, v_{uij}$  potenciál az  $(i,j)$  pozícióban lévő sejt ún. belső állapota, kimenete és a sejthez tartozó bemenet.

♦  $I_{xu}(i,j,k,l), I_{xy}(i,j,k,l)$  vezérelt áramforrások. A következő egyenletek írják le működésüket:

$$I_{xu}(i,j,k,l) = B(i,j,k,l)v_{ukl}, \quad I_{xy}(i,j,k,l) = A(i,j,k,l)v_{ykl} \quad (11.1)$$



Itt  $A(i,j,k,l)$  a  $(k,l)$  pozíciójú sejthez tartozó bemenet  $(i,j)$  sejtre vonatkozó csatolásának súlya,  $B(i,j,k,l)$  pedig a  $(k,l)$  pozíciójú sejthez tartozó kimenet  $(i,j)$  sejtre vonatkozó visszacsatolásának súlya.

◆ I az adott sejtben alkalmazott konstans áram előfeszítés.

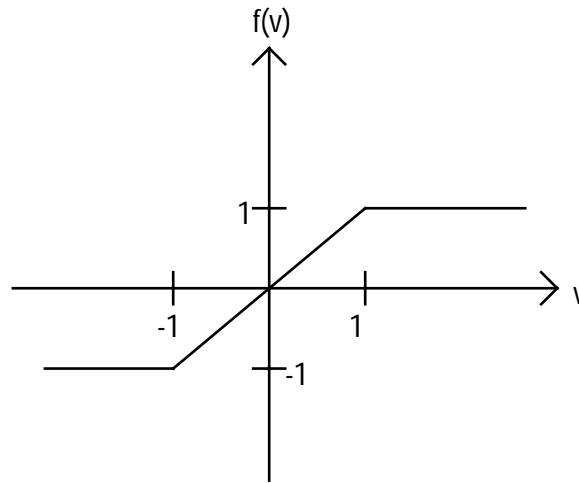
◆ A bemenetből a kimenetet az áramgenerátor alakítja ki az ellenállással.

$$I_{yx} = \frac{1}{2R_y} (|v_{xij} + 1| - |v_{xij} - 1|) \quad (11.2)$$

Így a sejt kimenete a következőképp alakul:

$$v_{yij} = f(v_{xij}) = \frac{1}{2} (|v_{xij} + 1| - |v_{xij} - 1|) \quad (11.3)$$

Az  $f$  függvény grafikonja a következő:



11.3.Ábra

Általános esetben a kimenetet a belső állapotból másféle nemlinearitás is előállíthatja.

Az ilyen egyszintű hálózat egy sejtjét a következő differenciálegyenlet írja le:

$$C \frac{dv_{xij}(t)}{dt} = \frac{-1}{R_x} v_{xij}(t) + \sum_{(k,l) \in N_r(i,j)} A(i,j,k,l) v_{ykl}(t) + \sum_{(k,l) \in N_r(i,j)} B(i,j,k,l) v_{ukl}(t) + I$$

$$v_{yij} = f(v_{xij})$$

(11.4)

$N_r$  arra utal, hogy a sejtekre csak  $r$  sugarú szomszédságuk hat. Gyakori a  $C=1$ ,  $R=1$  választás. Ekkor az alak leegyszerűsödik:

$$\frac{dv_{xij}(t)}{dt} = -v_{xij}(t) + \sum_{(k,l) \in N_r(i,j)} A(i,j,k,l) v_{ykl}(t) + \sum_{(k,l) \in N_r(i,j)} B(i,j,k,l) v_{ukl}(t) + I$$

(11.5)

Figyelembe véve, hogy egy sejtre csak  $r$  sugarú környezete hat:

$$\frac{dv_{xij}(t)}{dt} = -v_{xij}(t) + \sum_{\Delta i=-r}^r \sum_{\Delta j=-r}^r A(i, j, i + \Delta i, j + \Delta j) v_{ykl}(t) + \sum_{\Delta i=-r}^r \sum_{\Delta j=-r}^r B(i, j, i + \Delta i, j + \Delta j) v_{ukl}(t) + I \quad (11.6)$$

Az adott sejt egyenletében előforduló  $A$  és  $B$  együtthatókat szokás mátrixos alakban is megadni. Ekkor  $A$  és  $B$  mátrix  $(2r+1) \times (2r+1)$  méretű. Az  $A$  mátrix pl.  $r=1$  esetre a következőképp alakul:

$$\underline{A} = \begin{bmatrix} A(i, j, i-1, j-1) & A(i, j, i, j-1) & A(i, j, i+1, j-1) \\ A(i, j, i-1, j) & A(i, j, i, j) & A(i, j, i+1, j) \\ A(i, j, i-1, j+1) & A(i, j, i, j+1) & A(i, j, i+1, j+1) \end{bmatrix} \quad (11.7)$$

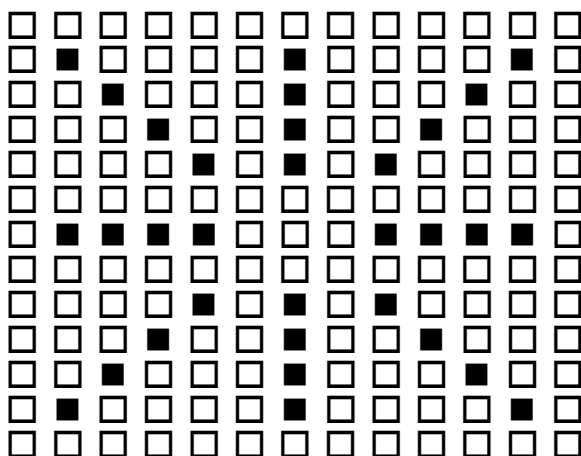
A  $B$  mátrix hasonlóan néz ki, csupán a képletben az  $A(i, j, k, l)$  függvény helyett a  $B(i, j, k, l)$  függvény szerepel.

A legtöbb esetben az egyrétegű hálózatot azonos sejtek alkotják, így minden sejtet ugyanaz az  $A$  és  $B$  mátrix, valamint  $I$  áramérték jellemez. Ekkor  $r=1$  környezettel való érintkezés esetén  $A$  és  $B$   $3 \times 3$ -mas, a hálózat  $3 \times 3 + 3 \times 3 + 1 = 19$  valós számmal jellemezhető. Ugyanezt  $r=2$  környezet esetén  $5 \times 5 + 5 \times 5 + 1 = 51$  valós számmal tehetjük meg. (Megjegyzem, hogy az általam létrehozott hálózatnál az  $A$ ,  $B$  és  $I$  értékek nem azonosak minden sejtre.)

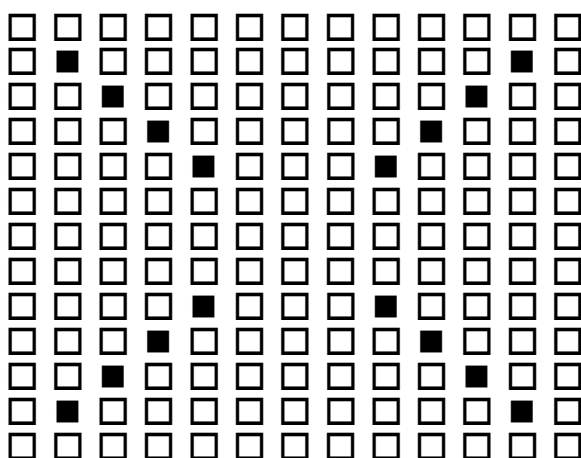
Példaként bemutatok egy egyszerű ún. template összeállítást, amely az átlókat detektálja. A bemenetre adott képből a kimenetre csak az átlók maradnak, a vízszintes vonalak eltűnnek. Itt a sejtek  $r=1$  környezetükkel állnak kapcsolatban, tehát  $A$  és  $B$   $3 \times 3$ -mas mátrix.

$$\underline{A} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \underline{B} = \begin{bmatrix} 0,25 & 0 & 0,25 \\ 0 & 0 & 0 \\ 0,25 & 0 & 0,25 \end{bmatrix}, I = -0,5 \quad (11.8)$$

A következő ábra a bemenetet mutatja és a hatására létrejövő kimenetet. (A fekete négyzet  $+1$ -et, a fehér  $-1$ -et jelent.)



11.4.a.Ábra



11.4.b.Ábra

## 11.2.A szimulátorprogram

A hálózat szimulálására Turbo Pascal nyelven számítógépprogramot készítettem, amely a celluláris neurális hálózat differenciálegyenletét 40x40-es sejtrendezés esetén az előrelépő Euler formulával oldja meg ( $R=1$ ,  $C=1$ ). A program neve FELULET.EXE. A hozzá tartozó paramétereket a FELULET.PAR szövegfájl tartalmazza. Szerkezete azért is érdekes, mert gyorsan áttekinthetővé teszi a szimuláció során felmerülő feladatokat. Ezt a bemeneti adatlapot mutatja a következő ábra. A dőltbetűs részeket kell a felhasználónak kitölteni.

\*\*\*Paraméter állomány a FELULET.EXE programhoz\*\*\*

A bemenet:0-két "cni" állomány, 1-egy "poz" állomány

*egész1*

A bementi képet megadó "cni" állomány:

*állománynév1*

A bementi kép maszkolását megadó "cni" állomány:

*állománynév2*

A bemeneti képet megadó "poz" állomány:

*állománynév3*

Új kép(0) vagy kép folytatása(1):

*egész2*

A folytatandó képet megadó "cni" állomány:

*állománynév4*

Az illesztettség mértéke(0-interpoláció):

*valós1*

Időlépés:

*valós2*

Iterációk száma(ha <0, futáskor kérdezi meg):

*egész3*

Hány iteráció után írjon ki hibainformációt:

*egész4*

A kimeneti kép "cni" állományának neve:

*állománynév5*

### 11.5.Ábra

Láthatóan a bemeneti (vagyis az ismert pontokat tartalmazó) kép megadható két "cni" kiterjesztésű állománnyal. Ez a CNNM V5.3-mas

szimulátorral kompatibilis forma. A "cni" állomány úgy írja le az adott képet, hogy először magadja x és y irányú méretét, majd sorfolytonosan felsorolja az összes rácspontbeli értéket, vagyis esetünkben  $40 \times 40 = 1600$  darabot. Az első állomány a bemeneti képet adja, a második az ismert pontokat jelöli ki. Az adott rácsponthoz tartozó érték 1, ha a pont nem ismert, -1, ha ismert. Ez az ún. fekete-fehér maszkolás. (Bővebben lásd [10].)

A másik megoldás az általam kifejlesztett "poz" kiterjesztésű állomány. Ez az ismert pontok koordinátahármasait tartalmazza egy-egy sorban. Az x és y koordináta 1-től 40-ig terjedő egész, a z koordináta valós. A fájlt -1 zárja le.

Lehetőség van már megkezdett képen a szimuláció folytatására. Ekkor meg kell adni az előző szimuláció végeredményét tartalmazó képet egy "cni" kiterjesztésű állományban. Erre a funkcióra szükség van, hiszen 10000 iteráció 0,015-ös időlépéssel akár 40 percet is igénybe vehet (386DX, 40MHz), így számítani kell rá, hogy a kép teljes feldolgozása több részletben történik. Megjegyzem hogy ez a 10000 iteráció csupán  $10000 \times 0,015 = 150$  időegységnek felel meg.

A szimulátor ismeri az interpolációt. Ekkor az illesztettség mértékeként 0-át kell megadni. És ismeri az approximációt is. Ekkor az illesztettség mértékeként  $2\alpha$ -át kell megadni. A szimulátor maga állítja elő a template-eket.

Az időlépés a gyakorlatban 0,01-0,015 volt. 0,02-nél már divergenciát tapasztaltam.

Az iteráció során lehetőség van hibainformációt kérni. Ez az utolsó iterációhoz képesti legnagyobb változást mutató rácselem változásának kiírását jelenti, illetve az egyes rácselem változások négyzetösszegeinek gyökét. Ez utóbbi tulajdonképpen a  $\Delta y$  vektor abszolút értéke.

A kimeneti állomány "cni" kiterjesztésű. Megnézhető az EDCNI nevű, CNNM rendszerhez tartozó programmal. Illetve az általam készített CNITOMTL.EXE programmal olyan MATLAB rendszerbe való állomány hozható létre, amely kirajzolja a képhez tartozó felületet, mint hálóabrát. Ezt használtam a diplomamunka példáinak megrajzolásában. Egy POZTOCNI.EXE nevű program pedig képes a felületet megadó "poz" kiterjesztésű állományból létrehozni a felületet és az ismert pontok helyét megadó két "cni" kiterjesztésű állományt.

## **12.Irodalomjegyzék**

- 1.W. E. L. Grimson, An Implementation of a Computational Theory of Visual Surface Interpolation, Computer Vision, Graphics, and Image Processing 22, 1983, 39-69
- 2.W. E. L. Grimson, A computational theory of visual surface interpolation, Philos. Trans. R. Soc. London. Ser. B 298, 1982, 395-427
- 3.W.E.L.Grimson, A computer implementation of a theory of human stereo vision, Philos. Trans. R. Soc. London Ser. B 292, 1981, 217-253
- 4.Demetri Terzopoulos, Multilevel Computational Processes for Visual Surface Reconstruction, Computer Vision, Graphics, and Image Processing 24, 1983, 52-96
- 5.R. Courant and D. Hilbert, Methods of Mathematical Physics, Vol. I, Interscience, New York, 1983, 250-252
- 6.Dr.Fodor György, Elméleti elektrotechnika III., Tankönyvkiadó, Budapest, 1989, 174-176
- 7.Nemes László, Képrekonstrukció a konfokális mikroszkópiában CNN segítségével, Diplomamunka, Budapesti Műszaki Egyetem, 1993, 46-47
- 8.CNNM Multi-layer Cellural Neural Network Simulator, user's guide, version 5.3.X., 1994
- 9.Roska Tamás, Elektronikus operátorok-neurális áramkörök, Előadásjegyzet, Budapest, 1992
- 10.Stoyan Gisbert és Takó Galina, Numerikus módszerek, Budapest, 1993