

# Mediánszűrés alkalmazása algebrai rekonstrukciós módszerekben<sup>\*</sup>

Hantos Norbert, Balázs Péter

Szegedi Tudományegyetem, Képfeldolgozás és Számítógépes Grafika Tanszék  
nhantos@inf.u-szeged.hu, pbalazs@inf.u-szeged.hu

**Absztrakt.** Képrekonstrukció alatt egy objektum kétdimenziós szeleteinek előállítását értjük vetületeinek ismeretében. A feladat megoldására számos eljárás létezik, ám ha a vetületek száma korlátozott, az egyik legjobb megoldást az algebrai rekonstrukciós módszerek szolgáltatják, amelyek iteratívan, a problémából levezetett egyenletrendszer kielégítésével közelítik a megoldást. A módszer család hátránya, hogy kevés iteráció, illetve kevés rendelkezésre álló vetület esetén az eredménykép zajos. Jelen cikkben megmutatjuk, hogyan lehet csökkenteni a zaj mértékét az iterációs lépések között alkalmazott mediánszűréssel. Az új eljárást több különböző strukturális összetettségű képen, többféle paraméterezéssel teszteltük. Az algebrai módszerek közül elsősorban az ART és DART eljárásokat használtuk.

## 1. Bevezetés

A képrekonstrukció feladata egy kétdimenziós kép előállítása vetületeinek ismeretében. Bizonyos alkalmazásokban, mint például az elektronmikroszkópia [1] vagy a neutron tomográfia [3], a rekonstrukció csak kevés vetületből, kevés iterációval történhet, mivel a vetítés sugarak roncsolhatják a vizsgált objektumot. Ilyenkor a többszáz vetületet igénylő transzformáció-alapú rekonstrukciós eljárások – szűrt visszavetítés, inverz-Radon transzformáció – nem alkalmazhatók sikerrel. A probléma megoldását az algebrai módszerek használata jelentheti, amelyek a vetületek által meghatározott egyenletrendszert iteratívan próbálják megoldani. Az eljárások eredményeként kapott kép azonban sokszor túlságosan szemcsés, zajos lesz [7].

Jelen cikkben egy olyan módszert mutatunk be, amely folytonos illetve diszkrét algebrai rekonstrukciós eljárásokban az egyes iterációk során elvégzett mediánszűréssel finomít az eredményképen. A mediánszűrés növeli a kép homogenitását a fontos élek elmosása nélkül, így a rekonstruált kép hibája csökkenthető. A tesztelést egy saját fejlesztésű programmal hajtottuk végre, amelyben implementáltuk a klasszikus ART, illetve a 2007-ben ismertett DART eljárást is.

---

<sup>\*</sup> A cikk eredményei az alábbi publikációban jelentek meg: Hantos, N., Balázs, P.: Image Enhancement by Median Filters in Algebraic Reconstruction Methods: An Experimental Study. 6<sup>th</sup> *International Symposium on Visual Computing, 2010, Lecture Notes in Computer Science* (Springer) 6455/2010, 339–348.

Tesztképeink mesterségesen előállított, homogén régiókat tartalmazó képek voltak. A rekonstrukciók során az egyes eljárásokban explicit módon meghatározott gyakorisággal alkalmaztunk különböző méretű szűrőket. Az eredmények tükrében kijelenthető, hogy mind folytonos, mind diszkrét rekonstrukció esetén megfelelő szűrővel a szükséges iterációk, illetve vetületek száma hatékonyan csökkenthető ugyanolyan hibamértékű kép előállításához, csekély futásidő növekedés mellett.

A cikk felépítése a következő. A 2. fejezetben ismertetjük a rekonstrukciós probléma alapját. A 3. fejezetben részletezzük az általunk alkalmazott két fő algebrai rekonstrukciós módszert. A 4. fejezetben bemutatjuk, hogyan lehet szűrőket alkalmazni a rekonstrukció során. Az 5. fejezetben ismertetjük az eredményeinket. Végül a 6. fejezetben összegezzük a tapasztalatainkat és megfogalmazzuk a további terveinket.

## 2. A rekonstrukciós probléma

A tomográfia alapfeladata kétdimenziós képek előállítása annak a vetületeik ismeretében. Folytonos esetben ez azt jelenti, hogy meg kell határoznunk az  $f(x, y) : \mathbb{R}^2 \rightarrow \mathbb{R}$  függvényt (a képet) a Radon-transzformáció által megadott vonal menti integrálokból, ahol  $f$ -nek egy adott  $\theta$  szöggel vett párhuzamos vetülete az

$$[Rf](s, \theta) = \int_{-\infty}^{\infty} f(s \cos \theta - u \sin \theta, s \sin \theta + u \cos \theta) du \quad (1)$$

képlettel adott, ahol  $s$  és  $u$  a  $\theta$  szöggel elforgatott koordinátarendszer változóit jelöli (lásd 1. ábra bal oldala).

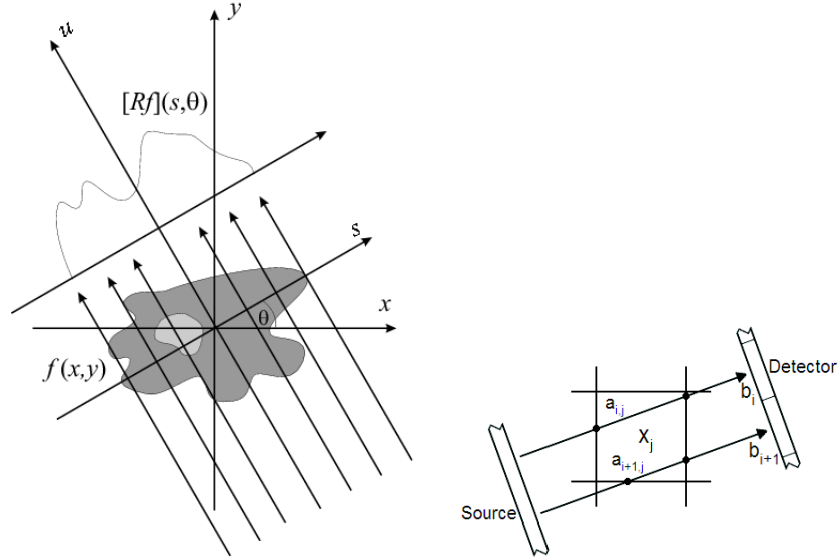
Bár a rekonstrukciós feladat az összes vetület ismeretében egyértelműen megoldható, a gyakorlatban csak véges sok vetületet képezhetünk, a mérési eredményeket pedig szinte mindig zaj terheli. A képet pixelek halmazával – négyzet-rácshálóval vagy mátrixszal – reprezentáljuk, így a rekonstrukciós problémát az alábbi módon írhatjuk le: egy-egy vetületi sugár legyen egy egyenes vonal, ami keresztülhalad a képmátrixon. A mátrix elemei a kép adott részének elnyelési együtthatóját reprezentálják. A sugár mentén mért elnyelődés ekkor kifejezhető az egyes érintett pixelek súlyozott összegeként. Így a rekonstrukciós probléma átírható egy egyenletrendszer megoldásának keresésére:

$$\mathbf{Ax} = \mathbf{b}, \mathbf{A} = (a_{i,j})_{n^2 \times m} \in \mathbb{R}^{n^2 \times m}, \mathbf{x} \in \mathbb{R}^{n^2}, \mathbf{b} \in \mathbb{R}^m, \quad (2)$$

ahol a rekonstruálni kívánt kép  $n \times n$ -es méretű,  $m$  a vetületi sugarak száma,  $a_{i,j}$  az  $i$ -edik sugár által érintett  $j$ -edik mátrixelem súlya,  $b_i$  pedig az  $i$ -edik sugáron mért elnyelődés (lásd 1. ábra jobb oldala).

## 3. Algebrai rekonstrukciós módszerek

A (2) egyenletrendszer közvetlen módon történő megoldása bonyolult és számításigényes. A problémát tovább nehezíti, hogy a rendelkezésre álló kevés vetület



**1. ábra:** Balra:  $f(x, y)$  függvény és egy  $\theta$  szögű vetülete. Jobbra: a rekonstrukció geometriája: az ismeretlen értékű  $x_j$  pixel és két, őt metsző vetületi sugár.

miatt az egyenletrendszer általában nagy mértékben alulhatározott, a vetületeken jelentkező zaj pedig inkonzisztensé teheti az egyenletrendszert. Az algebrai rekonstrukciós módszerek ezeket a problémákat úgy próbálják meg áthidalni, hogy iteratíván közelítenek egy olyan megoldást, melynek a hibája minimális. Az iterációs közelítésre több különböző módszer létezik, az egyik legalapvetőbb az úgynevezett Algebrai Rekonstrukciós Technika (Algebraic Reconstruction Technique, ART) [6].

Az ART az iteráció egy lépésében az aktuális képen minden vetületi egyenes mentén a mért hibát sorban visszavetíti az adott egyenes által érintett pixelekre. Vagyis a (2) egyenletrendszer egy aktuális megoldásának tekintett  $\mathbf{x}$  értékén úgy változtat, hogy az adott vetületi sugár által reprezentált egyenletben az eredetileg mért  $b_i$  és az aktuális  $\mathbf{x}$  értékek szerinti vetületi különbséget szétosztja az egyenlet  $\mathbf{x}$  változóin. Formálisan a hiba mértéke az  $i$ -edik vetületi sugár esetén:

$$\Delta = a_{i,1}x_1 + a_{i,2}x_2 + \dots + a_{i,n^2}x_{n^2} - b_i . \tag{3}$$

Az  $i$ -edik sugár szerinti összsúly  $W = \sum_{j=1}^{n^2} a_{ij}$ . A hiba elosztása a sugár által érintett pixeleken az adott súly szerint történik. Tehát az új  $\mathbf{x}'$  értékek:

$$x'_1 = x_1 + a_{i,1} \frac{\Delta}{W}, \dots, x'_{n^2} = x_{n^2} + a_{i,n^2} \frac{\Delta}{W} . \tag{4}$$

Ebben az esetben az adott sugár által érintett képpontokon a hiba mértéke 0 lesz, ám más sugarak esetén a hiba növekedhet. Az algoritmus egy iterációs

lépésben ezt az eljárást sorban végrehajtja minden sugáron, majd az iterációt előről kezdi. Az ART-ot és tulajdonságait részletesen a [7] könyv tárgyalja.

Diszkrét kép rekonstrukciója esetén (*diszkrét tomográfia*) feltételezzük, hogy az  $f$  képfüggvény értékészlete véges, kis elemszámú halmaz. Szélsőséges esetben  $f$  csak két értéket, 1-et és 0-át vehet fel, ami egy adott anyag jelenlétét vagy hiányát jelzi. Ebben az esetben *bináris tomográfia*ról beszélünk. Ha tudjuk, hogy egy rekonstruálni kívánt diszkrét (bináris) kép milyen szürkeintenzitási értékeket tartalmazhat, akkor egy kellően jó eredménykép előállításához már kevés vetület is elegendő lehet. A diszkrét tomográfia elméleti háttéréről és rekonstrukciós eljárásairól a [8, 9] művek adnak részletesebb felvilágosítást.

Egy hatékony diszkrét rekonstrukciós eljárás az ART egyik variánsa, a Diszkrét Algebrai Rekonstrukciós Technika (Discrete Algebraic Reconstruction Technique, DART) [2]. Az eljárás alapja, hogy egy tetszőleges folytonos rekonstrukciós módszer eredményképén vett küszöbölés megközelítőleg jó eredményt ad a diszkrét feladat megoldására és általában csak a határvonalon pontatlan. A DART ezért egy kezdeti ART rekonstrukció után a határvonalon finomít iterációról iterációra. A DART-ot és tulajdonságait részletesebben a [2] cikk tárgyalja. Az eljárás fő lépései a következők:

1. Az ART valamely változatával egy folytonos, kezdeti rekonstrukciót végzünk.
2. Az aktuális  $\mathbf{x}_{act}$  képet az ismert diszkrét értékek szerint küszöböljük. Az így kapott aktuális diszkrét kép  $\mathbf{x}'$ .
3. Meghatározzuk a határvonalhoz nem tartozó pixelek  $S$  halmazát  $\mathbf{x}'$ -ből.
4. Előállítjuk az  $\mathbf{x}''$  új folytonos aktuális képet oly módon, hogy ha egy pixel benne van az  $S$  halmazban, akkor az értéke a küszöbölt érték, ha nem, akkor az értéke az  $\mathbf{x}_{act}$ -beli érték.
5. Végrehajtunk egy ART iterációs lépést az  $\mathbf{x}''$  képen úgy, hogy az  $S$ -beli pontokat rögzítjük.
6. Simítást végzünk a határpontokon. Így kapunk egy új  $\mathbf{x}''$ -t.
7. Ha a megállási feltétel teljesül, egy végső küszöbölés után befejezzük az eljárást, egyébként folytatjuk a 2. lépéstől úgy, hogy  $\mathbf{x}_{act} = \mathbf{x}''$ .

#### 4. Mediánszűrés a rekonstrukcióban

Ha a vetületek száma kevés, az algebrai rekonstrukciós eljárások által generált eredménykép sokszor só-bors jellegű zajjal terhelt [7]. Célunk annak vizsgálata, hogyan lehet a zaj mértékét csökkenteni a rekonstrukció iterációs lépései közben alkalmazott szűrővel. A zaj jellege miatt célszerű mediánszűrőt választani a zajszűréshez [5]. A mediánszűrő a kép minden pixeljét lecseréli a pixel környezetében található pixelek értékeinek a mediánjára. Pontosabban kifejezve, ha  $f(x, y)$  a kép  $(x, y)$  koordinátájában található értéket jelöli, akkor akkor az új  $f'(x, y)$  érték a következő:

$$f'(x, y) = \underset{-k \leq u, v \leq k}{\text{med}} \{f(x + u, y + v)\}, \quad (5)$$

vagyis a mediánt egy  $(2k+1) \times (2k+1)$ -es méretű ablakban található értékekből számíthatjuk, ez a mediánszűrő mérete. Egy adott halmaz mediánja a halmaz elemeinek nagyság szerint sorbarendezett elemei közül a középső lesz.

A mediánszűrő hatására a zaj a képen nagymértékben csökken, ugyanakkor a megkívánt vetületektől való eltérés növekedhet. Mivel a hibát minimalizálni szeretnénk, ezért csak minden  $l$ -edik iterációban alkalmazunk mediánszűrést, ahol  $l$  egy paraméter.

DART rekonstrukció esetén nem csak a kezdeti ART rekonstrukcióban, hanem a DART iterációk során is alkalmazhatunk szűrést. A tesztelés során erre mint *belső szűrés* fogunk hivatkozni.

## 5. Tapasztalati eredmények

### 5.1. Az implementáció

Ahhoz, hogy a mediánszűrők hatását tesztelni tudjuk, szükségünk volt egy általános keretrendszerre. Az implementáció során feltételeztük, hogy a vetületi sugarak egymástól mindig egységnyi távolságra vannak, párhuzamosak, egyenesek és vastagság nélküliek. A gyors számításhoz azt is feltettük, hogy a képpontok súlya az egyes vetületi sugarak esetén 1 vagy 0 attól függően, hogy az adott sugár érintette az adott képpontot vagy sem. Vagyis a (2) egyenletrendszerben  $\mathbf{A}$  egy bináris mátrix. Bár a többértékű súlyokkal pontosabb rekonstrukció végezhető el, esetünkben az eljárások összevetéséhez elegendőnek találtuk a bináris súlyok alkalmazását, így lehetővé vált, hogy az egyes vetületi sugarak által érintett pixeleket a Bresenham vonalrajzoló algoritmus segítségével számítsuk [4]. A végső implementáció tartalmazza mind az ART, mind a DART rekonstrukciós eljárást. Az eljárásoknak megadhatjuk a tesztelni kívánt képet, és a vetületek számát. Adott vetületszám esetén a vetületeket egységesen osztjuk el a vízszintes irányú vetületből kiindulva. Ezen felül beállíthatjuk, alkalmazunk-e szűrőt, és ha igen, milyen mérettel és gyakorisággal. DART rekonstrukció esetén még megadhatjuk, hogy *belső szűrést* is akarunk-e végezni. A mediánszűréshez egy előre implementált, gyors algoritmust használtunk [10]. Az eljárás akkor áll meg, amikor elértünk egy előre megadott iterációs számot.

Az implementáció Windows 7 operációs rendszer alatt készült, a tesztelés egy 1.5 GHz-es, Intel Core 2 Duo T2520 processzorral és 2 GB RAM-mal rendelkező gépen zajlott.

### 5.2. A tesztképek

Bináris esetben összesen 6 különböző strukturális összetettségű tesztképet vizsgáltunk. Két képet közülük, a *Simple*-t és a *Cylinders*-t – lásd a 2. ábrán az első és a második kép – a [2] cikkből vettük, hogy összehasonlíthassuk a standard DART algoritmus által számított eredményeket a saját eredményeinkkel. A két kép mérete  $512 \times 512$ -es.

Folytonos rekonstrukcióhoz a 2. ábrán látható harmadik és negyedik képet használtuk. A *Phantom1* kép egyszerűbb, homogén területeket tartalmazó, négy

különböző intenzitási értéket tartalmazó kép. A Phantom2 kép komplexebb, folytonos átmeneteket tartalmazó kép. Mindkét kép mérete  $256 \times 256$ .



**2. ábra:** A rekonstrukcióhoz használt tesztképek. Sorrendben: a Simple és a Cylinders képeket bináris, a Phantom1 és Phantom2 képeket folytonos rekonstrukcióhoz használtuk.

### 5.3. Mérési eredmények

A szűrő alkalmazásával két különböző célt próbáltunk elérni. Az első esetben csökkenteni szerettük volna a szükséges iterációk számát adott zajszint megengedése mellett, ezzel gyorsítva a rekonstrukciót. A másik esetben adott iterációszám mellett megpróbáltunk minél kisebb zajmértékű eredményképet produkálni. Ebben az esetben előfordulhat, hogy kevesebb vetület is elegendő egy közel ugyanolyan minőségű kép előállításához, ami kulcsfontosságú a rekonstrukció gyakorlati alkalmazásaiban. Bináris rekonstrukció során a hiba mértékét a relatív átlagos hiba szerint mértük (Relative Mean Error, RME), az alábbi módon:

$$RME = \frac{\sum_{i=1}^{n^2} (p_i - p'_i)^2}{|p_o|}, \quad (6)$$

ahol  $|p_o|$  a fehér pontok – az objektumpontok – száma az eredeti képen,  $p_i$  az eredeti,  $p'_i$  az eredménykép  $i$ -edik pontja,  $n^2$  a kép pontjainak a száma.

Folytonos esetben a hiba mértékét a négyzetes hibaösszeggel mértük:

$$ERR = \frac{\sum_{i=1}^{n^2} (p_i - p'_i)^2}{n^2}, \quad (7)$$

ahol a jelölések megegyeznek az előző képlet jelöléseivel.

A Simple kép esetén először szűrő nélkül rekonstruáltunk a [2] cikkben használt paraméterezéssel. Akárcsak a többi binárisan rekonstruált kép esetében, a kezdő ART rekonstrukció iterációszáma 10 volt. A [2] cikkben írtakhoz hasonlóan mi is úgy találtuk, hogy körülbelül 100 DART iteráció elegendő egy rekonstrukcióhoz, efölött a javulás már elhanyagolható. A 1. táblázat első három sora az eredeti paraméterezés szerinti rekonstrukció eredményeit mutatja. A

táblázat többi sorában látható, hogy 4 vetület esetén belső szűrő használata nélkül nem érhető el számottevő javulás, ám belső szűrővel az eredménykép hibája jelentősen csökkent. A legjobb szűrőparaméternek a 4 iterációnként használt  $11 \times 11$ -es méretű szűrőt találtuk.

A második esetben megpróbáltuk az iterációs számot csökkenteni, ehhez a korábban jónak talált  $4/11 \times 11$ -es (4 iterációnként alkalmazott  $11 \times 11$ -es méretű) szűrőt használtuk, és 5 vetületből rekonstruáltunk. A 2. táblázatból látszik, hogy nincs jelentős különbség a szűrővel használt és a szűrő nélküli rekonstrukcióban.

Vetületek száma	DART iterációk száma	Szűrő gyakoriság / méret	Belső szűrés	Számítási idő (s)	Hiba
4	120	nincs	nincs	13.1	0.0478821
5	110	nincs	nincs	13.8	0.0035362
6	90	nincs	nincs	12.3	0.0033417
4	120	3 / $7 \times 7$	nincs	13.3	0.0380249
4	120	3 / $7 \times 7$	van	16.7	0.0079346
4	120	2 / $7 \times 7$	van	18.7	0.0053291
4	120	3 / $9 \times 9$	van	15.5	0.0037918
4	120	4 / $11 \times 11$	van	15.6	0.0029449
4	120	5 / $13 \times 13$	van	15.5	0.0039520
4	120	5 / $11 \times 11$	van	15.1	0.0039177

1. táblázat: Simple kép: Rekonstrukció kevesebb vetületből

DART iterációk száma	Szűrő gyakoriság / méret	Belső szűrés	Számítási idő (s)	Hiba
30	nincs	nincs	4.2	0.0034981
20	nincs	nincs	3.1	0.0046921
10	nincs	nincs	1.9	0.0115547
8	nincs	nincs	1.6	0.0136719
5	nincs	nincs	1.3	0.0175209
3	nincs	nincs	1.1	0.0218163
30	4 / $11 \times 11$	van	5.3	0.0031242
20	4 / $11 \times 11$	van	3.4	0.0031166
10	4 / $11 \times 11$	van	2.4	0.0088692
8	4 / $11 \times 11$	van	2.1	0.0144691
5	4 / $11 \times 11$	van	1.7	0.0176468
3	4 / $11 \times 11$	van	1.4	0.0215073

2. táblázat: Simple kép: Rekonstrukció kevesebb iterációval

A *Cylinders* képre, amely topológiailag igencsak különbözik az előző képtől, teljesen más eredményt kaptunk. A [2] cikk szerint legalább 10 irányból kell vetületeket venni, hogy elfogadható eredményképet kapjunk. Még szűrővel sem sikerült lecsökkentenünk a szükséges vetületek számát jelentős hibanövekedés nélkül (lásd 3. táblázat). Ellenben amikor megpróbáltuk csökkenteni az iterációk számát 10 vetület mentén vett rekonstrukció szerint, úgy találtuk, hogy már jóval kevesebb iteráció is elegendő lehet adott hibamértékű eredménykép eléréséhez, a futásidő csekély növekedése mellett. A  $2/9 \times 9$ -es paraméterű szűrő szerinti eredmények a 4. táblázatban láthatóak.

Vetületek száma	DART iterációk száma	Szűrő gyakoriság / méret	Belső szűrés	Számítási idő (s)	Hiba
9	130	nincs	nincs	24.6	0.0789070
10	110	nincs	nincs	19.9	0.0089111
11	120	nincs	nincs	24.5	0.0091553
9	130	3 / 7x7	van	27.6	0.0756454
10	110	3 / 7x7	van	23.2	0.0093765
11	120	3 / 7x7	van	28.1	0.0096245
9	130	8 / 11x11	van	25.5	0.0719452
9	130	11 / 9x9	van	25.4	0.0619812
9	130	9 / 9x9	van	25.5	0.0749626

3. táblázat: *Cylinders* kép: Rekonstrukció kevesebb vetületből

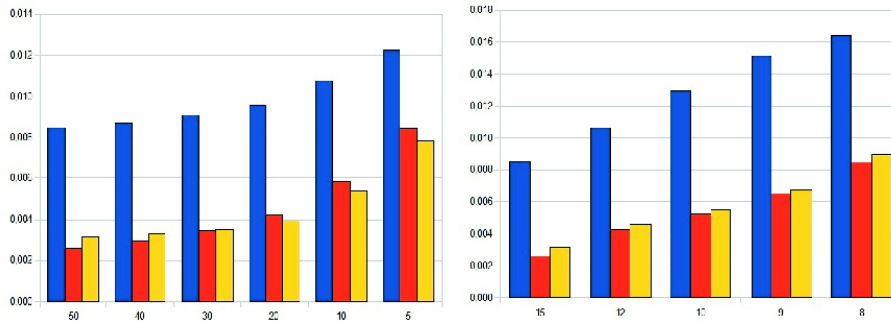
DART iterációk száma	Szűrő gyakoriság / méret	Belső szűrés	Számítási idő (s)	Hiba
80	nincs	nincs	15.1	0.0094261
70	nincs	nincs	12.9	0.0112724
60	nincs	nincs	11.5	0.0142479
50	nincs	nincs	9.7	0.0174675
30	nincs	nincs	6.4	0.0240402
20	nincs	nincs	4.6	0.0283852
80	2 / 9x9	van	18.1	0.0080528
70	2 / 9x9	van	16.2	0.0081215
60	2 / 9x9	van	14.1	0.0080185
50	2 / 9x9	van	11.9	0.0080452
30	2 / 9x9	van	7.9	0.0080528
20	2 / 9x9	van	5.9	0.0084724

4. táblázat: *Cylinders* kép: Rekonstrukció kevesebb iterációval



Az eljárásunkat további képekre is leteszteltük, a rekonstrukciós eredmények hasonló viselkedést mutattak: vagy a szükséges vetületek száma, vagy a szükséges iterációk száma jelentősen csökkenthető volt egy megfelelő paraméterezésű szűrő alkalmazásával. Sajnos úgy találtuk, hogy a szűrő paraméterezése nagyban függ a kép struktúrájától.

Folytonos rekonstrukciós esetben az eredményekről ismét mást mondhatunk. A 3. ábra diagramjai mutatják a Phantom1 rekonstrukciós eredményeit. A bal oldali diagram esetében 15 vetületből rekonstruáltunk, és az rekonstrukcióhoz használt iterációk számát csökkentettük 50-ről 5-re. A jobb oldali diagram esetében az iterációszámot 50-re rögzítettük, és a vetületek számát csökkentettük 16-ról 8-ra. Mindkét esetben két különböző szűrő szerinti rekonstrukciót is megvizsgáltuk.

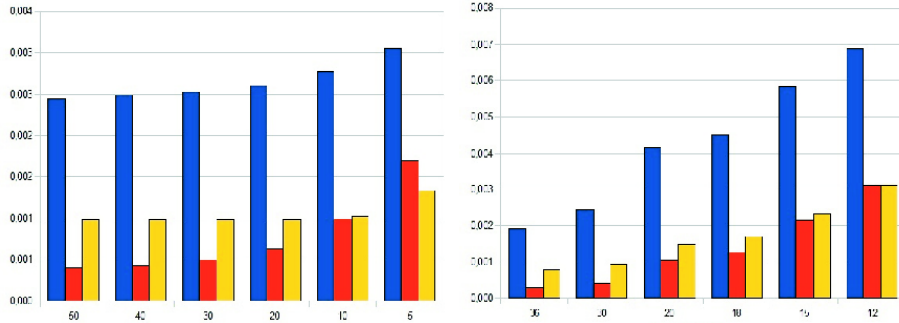


**3. ábra:** A Phantom1 kép rekonstrukciós eredményei az iterációk számának függvényében (bal), illetve a vetületek számának függvényében (jobb), abban az esetben, ha nincs szűrő (kék), ha a szűrő  $3/7 \times 7$ -es (piros), ha a szűrő  $2/11 \times 11$ -es (sárga).

A Phantom2 jóval bonyolultabb kép, ezért a rekonstrukcióhoz használt vetületek számát is emelnünk kellett. Az eredmények mégis hasonlóak a Phantom1 esetében kapott eredményekhez (lásd a 4. ábra diagramjait). Az első esetben 30 vetülettel dolgoztunk, az iterációk számát 50-től 5-ig vizsgáltuk. A második esetben 50-re rögzítettük az iterációk számát, a vetületek számát 16-tól 8-ig vizsgáltuk.

## 6. Összefoglalás és további tervek

Mivel az alap rekonstrukciós eljárások eredményképei általában só-bors zajjal terheltek, mediánszűrőt választottunk arra, hogy az iterációs lépések között javítsunk a képek homogenitásán. A mediánszűrők hatékonynak bizonyultak mind a diszkrét, mind a folytonos rekonstrukciós esetekben a rekonstrukció hibájának csökkentésére. Tapasztalataink szerint egy megfelelően választott szűrő lecsökkenti a rekonstrukcióhoz szükséges vetületek számát és/vagy a szükséges



**4. ábra:** A Phantom2 kép rekonstrukciós eredményei az iterációk számának függvényében (bal), illetve a vetületek számának függvényében (jobb), abban az esetben, ha nincs szűrő (kék), ha a szűrő  $3/5 \times 5$ -ös (piros), ha a szűrő  $2/13 \times 13$ -as (sárga).

iterációk számát, hatékonyabbá téve az eljárást. Sajnos úgy találtuk, hogy a megfelelő szűrő megválasztása nagyban függ a képtől, automatikus meghatározása nehéz lehet. További terveink között szerepel az automatikus szűrőparaméterezés megkeresése különböző gépi tanulási módszerek – például neuronhálózatok – vizsgálatával, kizárólag a vetületekből kinyerhető információ felhasználásával.

### Köszönetnyilvánítás

A kutatást részben a Magyar Nemzeti Fejlesztési Ügynökség TÁMOP-4.2.2/08/1/2008-0008 és TÁMOP-4.2.1/B-09/1/KONV-2010-0005 pályázatai valamint a Magyar Tudományos Akadémia Bolyai János Kutatási Ösztöndíja támogatta.

### Irodalom

1. K.J. Batenburg, S. Bals, J. Sijbers, C. Kuebel, P.A. Midgley, J.C. Hernandez, U. Kaiser, E.R. Encina, E.A. Coronado, G. Van Tendeloo, 3D imaging of nanomaterials by discrete tomography, *Ultramicroscopy*, **109(6)** (2009), 730–740.
2. K.J. Batenburg, J. Sijbers, DART: A fast heuristic algebraic reconstruction algorithm for discrete tomography, *Proceedings of the IEEE International Conference on Image Processing (ICIP), San Antonio, Texas, USA* **4** (2007), 133–136.
3. J. Baumann, Z. Kiss, S. Krimmel, A. Kuba, A. Nagy, L. Rodek, B. Schillinger, J. Stephan, Discrete tomography methods for nondestructive testing, *Applied and Numerical Harmonic Analysis*, Part III (2007), 303–331.
4. J.E. Bresenham, Algorithm for computer control of a digital plotter, *IBM Systems Journal*, **4** (1965), 25–30.
5. R.C. Gonzalez, R.E. Woods, *Digital Image Processing*, 2nd edition, Prentice Hall, New Jersey, 2002.
6. R. Gordon, R. Bender, G. T. Herman, Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography, *Journal of Theoretical Biology*, **29** (1970), 471–481.

7. G.T. Herman, *Fundamentals of Computerized Tomography: Image Reconstruction from Projections*, 2nd edition, Springer, 2009.
8. G.T. Herman, A. Kuba (Eds.), *Discrete Tomography: Foundations, Algorithms and Applications*, Birkhäuser, Boston, 1999.
9. G.T. Herman, A. Kuba (Eds.), *Advances in Discrete Tomography and Its Applications*, Birkhäuser, Boston, 2007.
10. S. Perreault, P. Héber, Median filtering in constant time, *IEEE Transactions on Image Processing*, **16** (2007), 2389–2394.