

# Pszedo-műhold repülőgépek képfeldolgozása

Mályusz Attila

Eötvös Loránd Tudományegyetem, Budapest

2021. december 16.

Pseudo-műhold repülőgép, mely

- 20 kilométer magasan
- több hónapig folyamatosan marad fenn
- önellátó → energia hatékony algoritmusok

Szárnyak és kamerák:

- A szárnyak modulokból épülnek fel.
- Egy-egy kamera a két szárnyon.
- Egy harmadik a repülőgép törzsének a végén.

Használata

- Internet továbbítás
- Mezőgazdasági tevékenységek kontrollálására
- Katasztrófa felmérés

## NASA Centurion [2]

- Első repülése 1998-ban
- Pár óráig repült.



## Facebook Aquila [1]

- Kétszer repült is: 2016, 2017
- 2018-ban leállították a projektet

A kész repülő méretei:

- Szárnyfesztávolsága Boeing 737-nek felel meg 34 méter.
- Tömege: 400 kg, míg egy Boeing 737-nek 42 tonna.



Kamerák által látott képekből szeretnék a lehető legjobb felbontású képet előállítani (Super-resolution).

Képeken kívül használt adatok a kamerák pozíciói. Ezek meghatározásához a következő adatokat használjuk fel:

- A rögzített képek.
- A repülőn elhelyezett szenzorok
- Harmadik kamera képeit.

Adott  $n$  db gyenge felbontású kép (LR) készítünk egy magas felbontású (HR) képet. Mátrixokkal megfogalmazva [4] [6]:

- LR képek  $Y_k$ , a keresett HR kép pedig  $X$ .
- Zajok  $V_k$
- Eltolás operátor  $F_k$
- Elmosódás is felép  $C$
- Down-sampling operator  $D$  (pixel szám csökkentés)

A reláció közöttük:  $Y_k = DCF_kX + V_k$ .

Feltesszük, hogy  $D$ -t,  $C$ -t és  $F_k$ -t tudjuk és  $C$  eltolás invariáns.

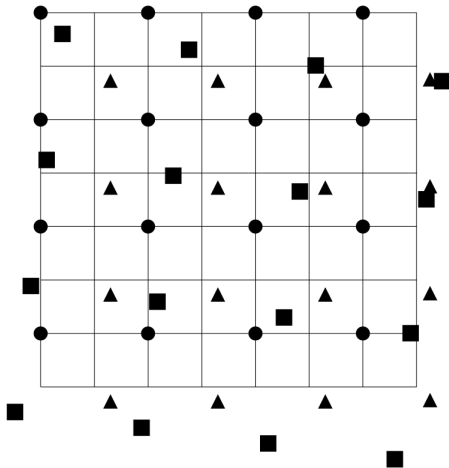


Figure 1: Low-resolution data on a high-resolution grid.

A nem neurális hálós technikák közül a [8] összefoglaló cikk alapján Wavelet interpolation bizonyult a legjobbnak.

- Egyrészt sokkal jobb eredményeket produkált másrészt gyorsabb is volt
- A wavelet függvényekre támaszkodik. [5]



Két lépésben készítjük a wavelet függvényeket.

Először veszünk egy  $\varphi(x) \in L^2(\mathbb{R})$  skála függvényt.

Ennek az eltolt és skálázott verzióival  $(\varphi(x))_{j,k} = 2^{\frac{j}{2}}\varphi(2^j x - k)$   $k, j \in \mathbb{Z}$  alakítunk ki altereket  $V_j = \overline{\text{Span}_k(\varphi_{j,k}(x))}$ . A skála függvénynek a következő négy feltételnek kell megfelelnie:

- a skála függvény ortogonális az egész eltoltjaira
- $\forall j \in \mathbb{Z}, V_j\text{-re} \subset V_{j+1}$ , ha  $f \in L^2(\mathbb{R})$
- $\forall j \in \mathbb{Z}, f \in V_j$ , akkor  $f \equiv 0$
- $V_\infty = L^2(\mathbb{R})$ .

Ekkor tudunk választani egy  $\psi(x)$  függvényt, melyre:

$\forall j \in \mathbb{Z} V_{j+1} = V_j \oplus W_j$ , ahol  $W_j = \overline{\text{Span}_k(\psi_{j,k}(x))}$ .

Ekkor  $\forall j \in \mathbb{Z} V_j \oplus W_j \oplus W_{j+1} \oplus \dots = L^2(\mathbb{R})$ .

Wavelet függvények haszna.

$f \in L^2(\mathbb{R})$ -re  $f(t) = \sum_{k \in \mathbb{Z}} a_{J,K} \varphi_{J,k}(t) + \sum_{j \geq J} \sum_{k \in \mathbb{Z}} b_{j,k} \psi_{j,k}(t)$ , ahol  
 $a_{J,K} = \int f(t) \varphi_{J,k}(t) dt$  és  $b_{j,k} = \int f(t) \psi_{j,k}(t) dt$ .

- $f$   $M$  helyen ismert
- 1D-s eset:  $f \in V_0$  és  $J \leq -1$ , ekkor  $V_0 = V_J \oplus \bigoplus_{j=J}^{-1} W_j$
- Kihhasználjuk, hogy előző alterek lineáris kombinációja és
- a szummák a lineáris kombinációban nem tartalmaznak végtelen sok elemet

Így mátrixos alakban írhatjuk fel a feladatot

$$f = G_J a_J + \sum_{j=J}^{-1} H_j b_j \text{ ahol:}$$

- $f = (f(t_i))_{i=0, \dots, P-1}$ ,  $a_J = (a_{J,k})_{k \in S_J}$
- $b_j = (b_{j,k}, k \in S_j)$
- $G_J = (\varphi_{K,k}(t_i))_{i=0, \dots, P-1}^{k \in S_J}$
- $H_j = ((\psi_{j,k}(t_i)))_{i=0, \dots, P-1}^{k \in S_j}$

- $a_J$  értékeket becsüljük meg  $\hat{a}_J$  regularized least squares módszerrel  $f \approx G_J a_J$ -ból.
- $b_J$ -t becsüljük meg  $g_J = f - G_J \hat{a}_J$ -ből.
- Ha több  $M$ -es mintánk is van, akkor  $f^{(i)} = G_J^{(i)} a_J$ -ra kell az előző módszert elvégeznünk.

- $f \in L^2(\mathbb{R}^2)$
- $2D$ -s skála függvény előáll a következőképpen  $\Phi(s, t) = \varphi(s)\varphi(t)$
- $f \approx (G_{J_s} \otimes G_{J_t})a_J$ , ahol  $\otimes$  a Kronecker szorzat és  $G_{J_s} \otimes G_{J_t}$



Facebook aquila.

[https://en.wikipedia.org/wiki/Facebook\\_Aquila](https://en.wikipedia.org/wiki/Facebook_Aquila).



Nasa centurion.

[https://en.wikipedia.org/wiki/NASA\\_Centurion](https://en.wikipedia.org/wiki/NASA_Centurion).



Tensor product.

<https://www.quantiki.org/wiki/tensor-product>.



M. Elad and A. Feuer.

Super-resolution reconstruction of an image.

*In Proceedings of 19th Convention of Electrical and Electronics Engineers in Israel*, pages 391–394, 1996.



R. C. Gonzalez and R. E. Woods.

*Digital Image Processing (3rd Edition)*.

Prentice-Hall, Inc., USA, 2006.



P. Milanfar.

*Super-Resolution Imaging.*  
CRC Press, 2011.



N. Nguyen and P. Milanfar.

A wavelet-based interpolation-restoration method for superresolution (wavelet superresolution).  
*Circuits, Systems and Signal Processing*, 19(4):321–338, Jul 2000.



D. Yang, Z. Li, Y. Xia, and Z. Chen.

Remote sensing image super-resolution: Challenges and approaches. In *2015 IEEE International Conference on Digital Signal Processing (DSP)*, pages 196–200, 2015.