

Párhuzamos képstílus átruházás konvolúciós neuronhálókkal

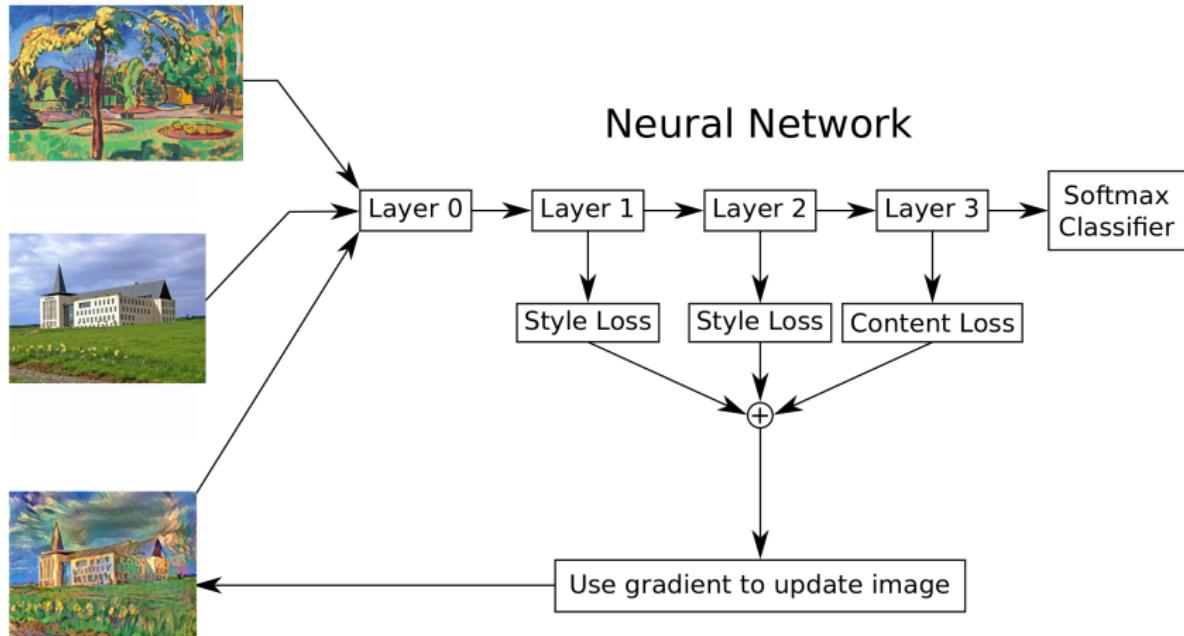
Szerző: Szilágyi Ervin

Témavezető: Dr. Iclánzan Dávid

Sapientia Eredélyi Magyar Tudományegyetem
Műszaki és Humántudományok kar
Szoftverfejlesztés szak

2017. július 6.

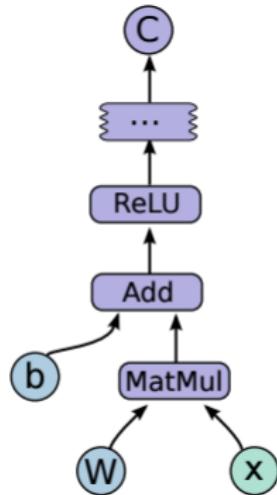
Mit értünk stílusátruházás alatt?



A dolgozat célja

- Grafikus felhasználói felülettel rendelkező alkalmazás fejlesztése
- Gépi tanulást (deep learning) alkalmazó rendszer tervezése és megvalósítása
- Híres magyar festők ismertebb műveinek művészeti stílusát átruházni képekre / mozgóképekre
- Párhuzamos gépi tanítási folyamat ami kihasználja a GPU által biztosított párhuzamosítási lehetőségeket

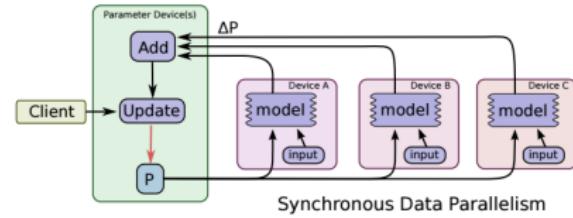
A Tensorflow könyvtár bemutatása



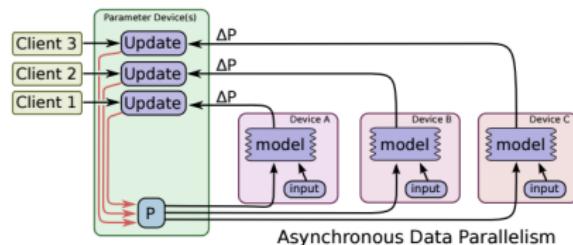
ábra. Tensorflow számítási gráf

Forrás: Google Research Team - *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*

Párhuzamos tanítás a Tensorflow segítségével

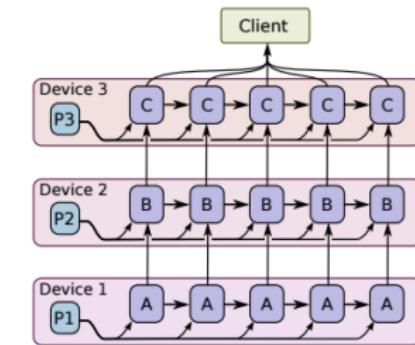


Synchronous Data Parallelism



Asynchronous Data Parallelism

(a) Adatpárhuzamos megközelítés



(b) Feladatpárhuzamos megközelítés

ábra. Párhuzamos tanítás

Forrás: Google Research Team - TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems

A rendszer tanítása

- Előre betanított neuronháló (VGG-19: 16 konvolúciós réteg, 5 pooling réteg);
- Statikus kép esetében külön tanításra kerül a bemeneti kép és a stílus kép is;
- Mozgókép esetében minden képkocka tanításra kerül;
- Temporális összefüggések a képkockák között.

A tanításhoz használt neuronháló

ConvNet Configuration						
A	A-LRN	B	C	D	E	
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers	
input (224×224 RGB image)						
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	
maxpool						
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	
maxpool						
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256	
maxpool						
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512	
maxpool						
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512	
maxpool						
FC-4096						
FC-4096						
FC-1000						
soft-max						

ábra. VGG-19 háló szerkezete

Forrás: Simonyan, K., Zisserman, A. - Very Deep Convolutional Networks for Large-Scale Image Recognition

A bemeneti kép tanítása

- A konvolúciós szűrök tartalmazzák a képek sajátosságait
- A bemeneti kép veszteségfüggvénye felírható mint:

$$L_{content}(\vec{x}, \vec{r}, I) = \frac{1}{2} \sum R_{ij}^I - W_{ij}^I \quad (1)$$

Ahol:

- \vec{x} - a bemeneti képet;
- R^I - az I-edik réteg válasza a bemeneti képre;
- W^I - az I-edik réteg válasza a fehér zaj bemenetre;
- \vec{r} - a kimeneti képet jelenti amit a rendszer generál a rétegek tulajdonságaiból.

Kiértékelt réteg: *conv4_2*

A stílus kép tanítása

A Gram mátrix ismertetése

- A Gram mátrix egy szorzatot jelen egy adott vektorhalmaz összes elemei között.
- Hogyha adott egy vektorhalmazunk $v_1 \dots v_n$, akkor a G Gram mátrixot a következő eljárás szerint határozzuk meg:

$$G_{ij} = v_i \cdot v_j \quad (2)$$

- A Gram mátrix ij pozíójában elhelyezkedő elem megadja, hogy egy adott réteg i -dik tulajdonsága mennyire teljesül a j -dik tulajdonság jelenlétében,

A stílus kép tanítása

A veszteségfüggvény meghatározása

Ha az I rétegnek N szűrője van, akkor felírható $G \in R^{N_I \times N_I}$ Gram mátrix, ahol:

$$G_{ij}^I = \sum_k F_{ik}^I \cdot F_{jk}^I \quad (3)$$

A veszteségfüggvény egyetlen rétegre felírható mint a fehér-zaj kép Gram mátrixa és a stílus kép Gram mátrixának átlagos négyzetes hibájaként:

$$E_I = \frac{1}{4N_I^2 M_I^2} \sum_{i,j} (G_{ij}^I - A_{ij})^2 \quad (4)$$

$$L_{style}(\vec{a}, \vec{x}) = \sum w_I E_I \quad (5)$$

Használt rétegek: `conv1_1`, `conv2_1`, `conv3_1`, `conv4_1`, `conv5_1`

A stilizált kép tisztítása

Total Variation Denoising

A stilizált képet eltoljuk X koordináta mentén egy pixellel, majd az Y koordináta mentén is eltoljuk egy pixellel.

$$L_{tv}(\vec{a}, \vec{x}) = \sum_{i,j} |(X_{ij} - A_{i+1,j})| + \sum_{i,j} |(X_{ij} - A_{ij+1})| \quad (6)$$

Végső veszteségfüggvénye függvény

$$L = L_{content} + L_{style} + L_{tv} \quad (7)$$

Optimalizációs algoritmus: ADAM (Adaptive Moment Estimation)
felhasználja a elsődleges, valamint a másodlagos gradiensek átlagát ahhoz,
hogy a veszteségfüggvényt minimalizálja.

Eredmény képek



(a) Eredit kép



(b) Stíluskép



(c) Eredmény

ábra. Csontváry - Tájkép naplemente idején



(a) Eredeti kép



(b) Stíluskép



(c) Eredmény

ábra. Munkácsi Mihály - Vihar a pusztán

Mozgókép tanítása

Naiv megközelítés

A videót feldaraboljuk képkockákra, majd az összes képkockára átruházzuk a stílust.

Előnyei:

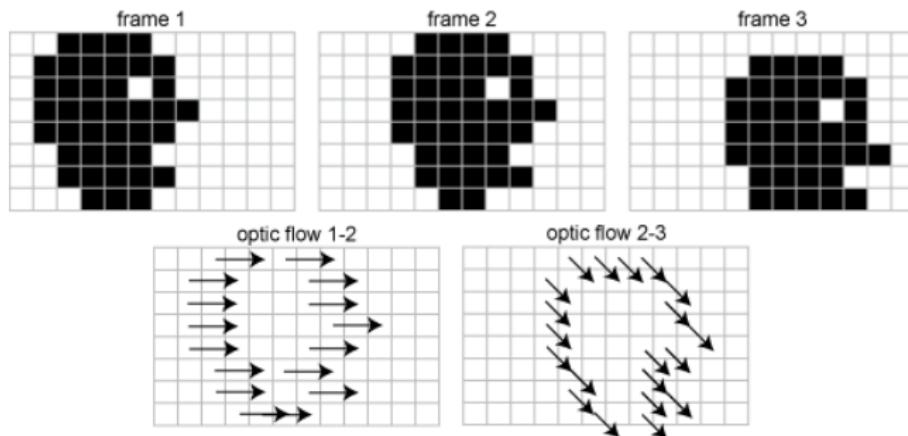
- Gyors.

Hátrányai:

- Nincs folyamatos átmenet a képkockák között
- "Flickering", "pop-ins"

Mozgókép tanítása

Optical flow bevezetése



Forrás: http://www.scholarpedia.org/article/Optic_flow

ábra. Optical flow szemléltetése

Mozgókép tanítása

A háló inicializálása

A cél:

- A mintázatok ugyanabba a pontba konvergáljanak
- "Flickering" jelenség elkerülése

Megoldás - a neuronhálót a következőképpnek inicializáljuk:

- Optical flow-t számolunk az előző stilizált képkocka és a mostani képkocka között
- Az előző stilizált képkockát torzítjuk az optical flow vektorai szerint

Mozgókép tanítása

Temporális összefüggések a képkockák között

Jelölje $w(u, v)$ az előre történő optical flow-t, valamint $\hat{w}(\hat{u}, \hat{v})$ a visszafele történő optical flow-t.

Az újonnan megjelenő objektumokra a következő egyenlőtlenség áll fenn:

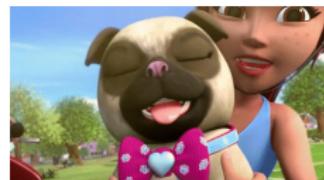
$$|w + \hat{w}|^2 > 0.01(|w|^2 + |\hat{w}|^2) + 0.5 \quad (8)$$

A mozgási határok pedig meghatározhatóak:

$$|\nabla u|^2 + |\nabla v|^2 > 0.01|w|^2 + 0.002 \quad (9)$$



(a) Frame00001



(b) Frame00002



(c) Forward optical flow

Mozgókép tanítása

Veszteségfüggvény meghatározása

$$L_{temporal}(x, w, c) = \frac{1}{D} \sum_{k=1}^D c_k \cdot (x_k - w_k)^2 \quad (10)$$

Ahol:

C - mátrix: mozgási határokra 1-est rakunk, a többi érték 0-ás

x - bemeneti kép

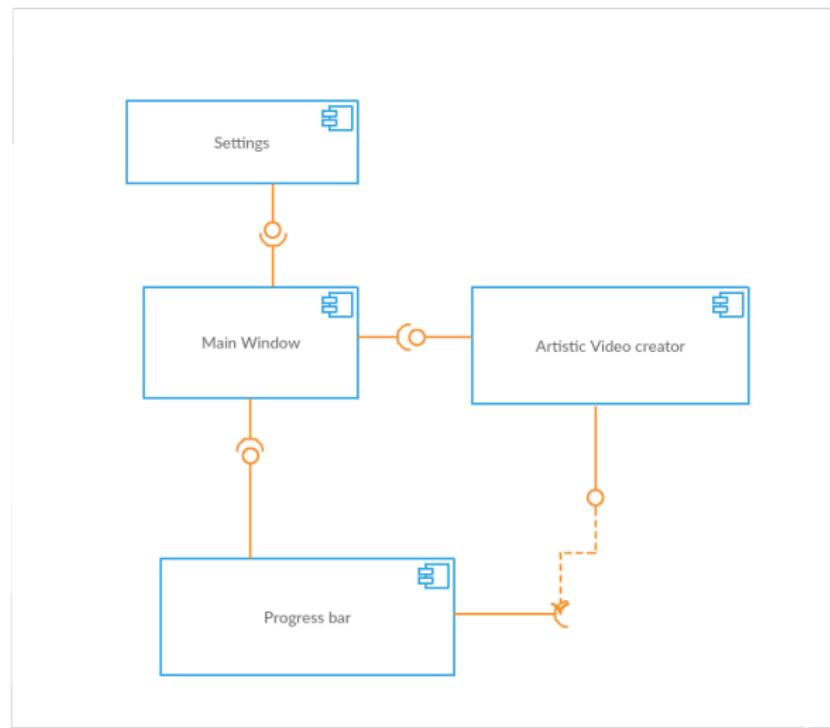
W - előző torzított kép

Végső tanítási függvény:

$$L = L_{content} + L_{style} + L_{tv} + L_{temporal} \quad (11)$$

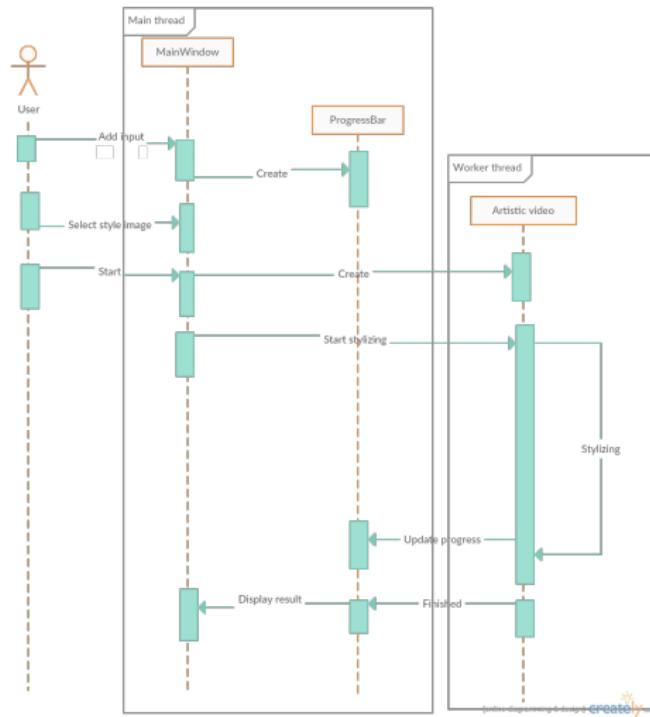
A szoftvet architektúrája

A szoftver komponensei



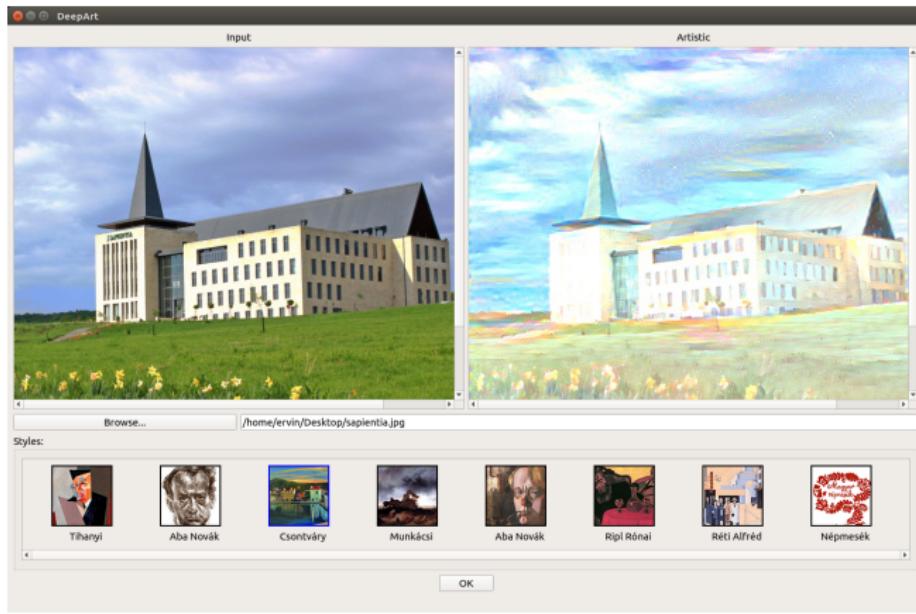
A szoftvet architektúrája

Többszás megvalósítás



A felhasználói felület

- Python
- PyQt5



Stílusképek



(d) Aba Novak
Vilmos - Önarckép



(e) Aba Novak
Vilmos - Selfportrait



(f) Kosztka Tivadar
- Traui tájkép
naplemente idején



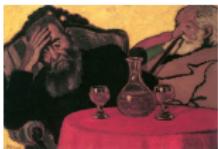
(g) Grunwald Béla -
Parkrészlet
Kecskeméten



(h) Munkácsi
Mihály - Vihar a
pusztán



(i) Réti Alfréd



(j) Ripl-Rónai József
- Apám és Piacsek...



(k) Tihanyi - Tzara
(l) Magyar
Népmesék



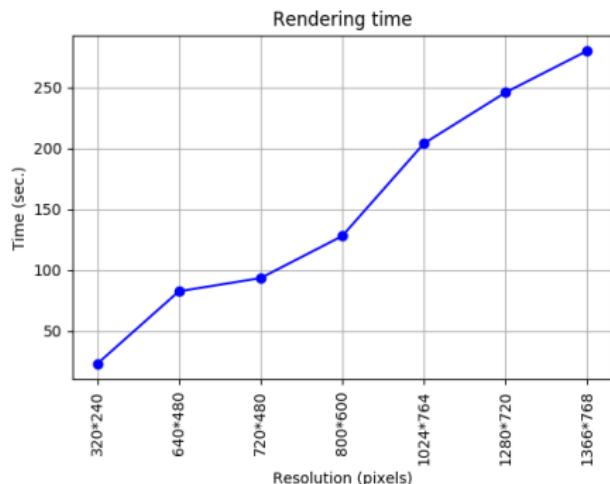
Mérések

A tesztkonfiguráció meghatározása

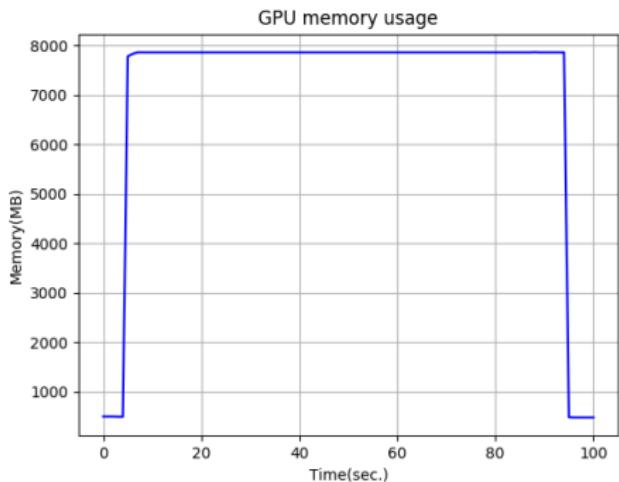
- **Alaplap:** MSI Z170A-G45 GAMING
- **Processzor:** Intel(R) Core(TM) Skylake i7-6700 ("non-K") CPU @ 3.40GHz (Turbo Boost: 3.90GHz)
- **Videokártya:** GIGABYTE GeForce GTX 1080 G1 GAMING 8GB DDR5X 256-bit
- **RAM Memória:** Corsair Vengeance LPX Black 32GB DDR4 3000MHz
- **Merev lemez:** HyperX Savage SSD, 240GB, 2.5", SATA III
- **Operációs rendszer:** Ubuntu 17.04 LTS

Egy képkockára történő stílusátruházási idő

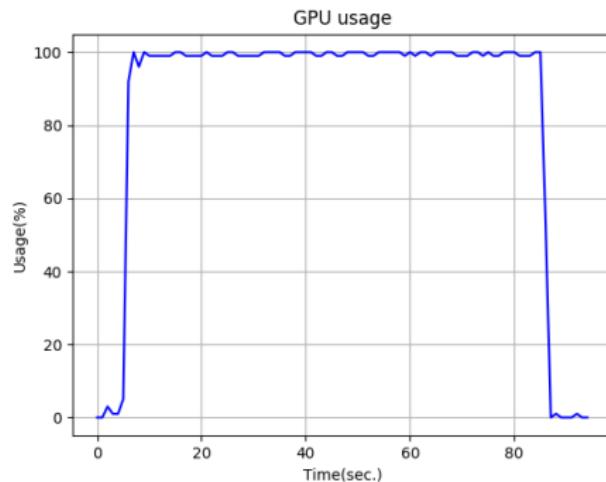
Felbontás(pixel)	Idő(másodperc)
320×240	23.126104
640×480	82.640347
720×480	93.487573
800×600	127.973041
1024×764	203.892024
1280×720	245.769916
1366×768	279.945757



Videókártya memória használata és kihasználtsága



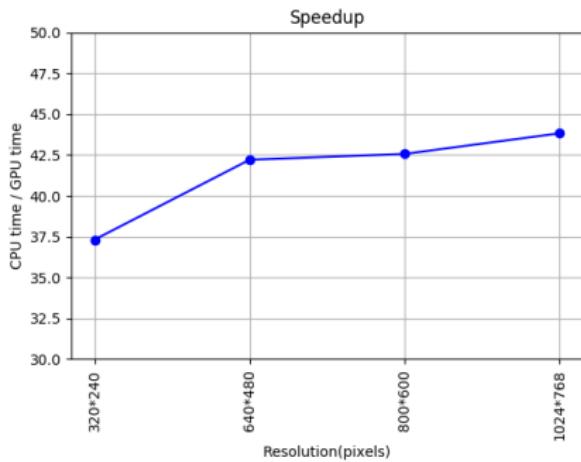
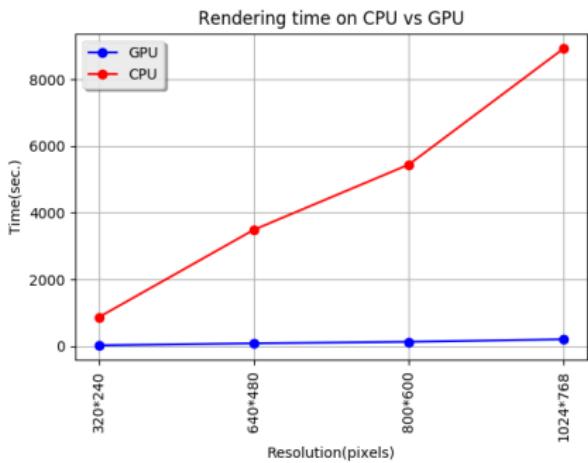
ábra. Memória használat



ábra. Kihasználtság

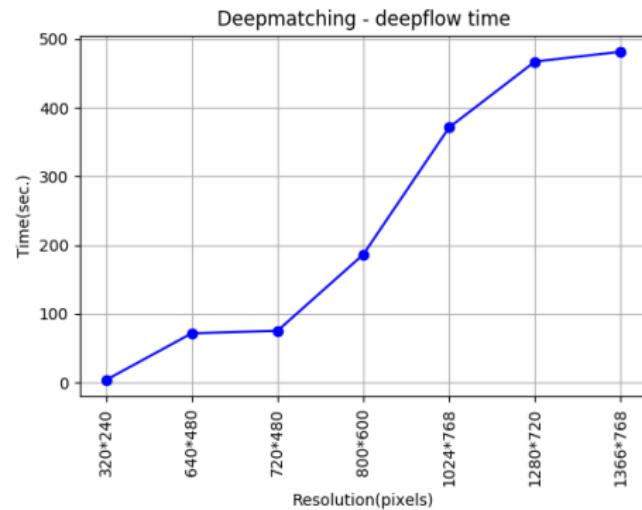
CPU - GPU összehasonlítás

Felbontás(pixel)	GPU Idő(sec.)	CPU Idő(sec.)	Gyorsulás
320×240	23.126104	862.980290	37.316285
640×480	82.640347	3487.931886	42.206162
800×600	127.973041	5446.849902	42.562479
1024×768	203.892024	8935.804122	43.826158



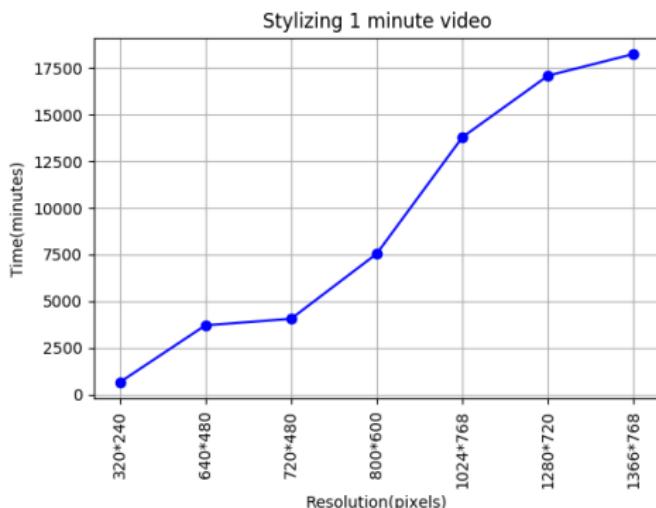
Az Optical flow időigénye

Felbontás(pixel)	Idő(másodperc)
320×240	3.700949
640×480	71.693549
720×480	75.388477
800×600	186.4160516
1024×764	371.140060
1280×720	466.996901
1366×768	481.25628



Egy perces videóra történő stílusátvitel

24 FPS, 60 másodperc, 1440 képkocka



ábra. Optical flow használatával



ábra. Optical flow használata nélkül

Összefoglaló

- Grafikus felhasználói felülettel rendelkező szoftver fejlesztése;
- Magyar festők híres műveinek stíluást alkalmazni minden napি képekre/mozgóképekre;
- Képkockák közötti temporális összefüggések kihasználása mozgóképek esetében;
- Tesztek, mérések elvégzése.

Továbbfejlesztési lehetőségek:

- Saját háló betanítása, más tanítási eljárás használata;
- Optical flow algoritmus gyorsítása (párhuzamosítás, más algoritmusok kipróbálása);
- Teljes Windows-os támogatás;
- Grafikus felület újratervezése, szépítése;
- Cloud alapú szolgáltatás készítése.

Köszönöm a figyelmet!

