*Bob Marley mozaikkép*

*7. Beadandó feladat Speciális algoritmusok tantárgyból*

Kalmár Dániel

Tomcsik Bence Tibor

**Feladat**

Bob Marley mozaikkép elkészítése. Készítsünk olyan algoritmust, amelynek a bemenete sok (akár különböző felbontású) kép, és egy célkép. A program készítse el a célképet, mint a többi képből álló mozaikképet. (Útmutató: először töltsük fel a képeket egy adatbázisba, amelyben tároljuk az adott kép legjellemzőbb színét, majd a célképet osszuk régiókra, és határozzuk meg egy régió legjellemzőbb színét (erre több módszer is jó lehet). A régió színének megfelelő képet keressük ki az adatbázisból (hatékonyan!), majd illesszük be kicsinyítve a megfelelő helyre.) Az egyszerűség kedvéért tekintsük a feladatot szürkeárnyalatos (grayscale) képekre. Pluszpontért készítsük el a programot színes képekre is. A programban legyen megadható, hogy a célkép hány sorra/oszlopra legyen felosztva.

**A feladat megoldásához használt algoritmus**

Vesszük a felhasználó által megadott sor- és oszlopadatokat és oszlop folytonosan végig megyünk az összes soron, közben vesszük az eredeti betöltött kép egy megfelelő méretű szeletét (amely a sorok és oszlopok számától függ), kiszámoljuk az átlagszínét, megkeressük ehhez az átlagszínhez legközelebb álló képet, lekicsinyítjük és rárajzoljuk az aktuális területre. Amennyiben a képarányok eltérnek, a kitöltő kép közepéből kivágunk egy megfelelő képarányú részt és azt használjuk csak fel, hogy ne torzuljon.

Átlagszámításnál az eredeti képet lekicsinyítjük egy adott méretre és annak a színátlagát számoljuk ki. A képek kicsinyítése azért gyorsabb megoldás, mert optimalizált és feltehetőleg hardver-gyorsított.

Azt, hogy melyik kép áll legközelebb az átlagszínhez nearest neighbor algoritmussal állapítjuk meg és lehetőségünk van két módszer használata közül választani:

* Tömbös:
* KD-tree: átlagos esetben.

Mérhető különbség alap esetben nincs köztük, mert a többi művelet meglehetősen lassú (kép betöltés, átalakítás). A különbség úgy jön elő, ha nagyon sok bemeneti kép van, amik felbontása egyezik a kitöltendő szeletek méretével.

**Forráskód**

A feladatot C# programozási nyelven oldottuk meg .NET 4.0 verzióval.

**Program használata**

A programot a marley.exe fájlal indíthatjuk el. Beállíthatjuk, hogy hány sorra és oszlopra bontsuk fel az eredeti képet. Betölthetjük a képeket amelyeket az algoritmushoz használunk fel, valamint a fő képet amin elvégezzük az algoritmust, amit a "Generate Mosaic" gombbal indíthatunk el. Használhatjuk színes képek esetén is.

**Eredmények**

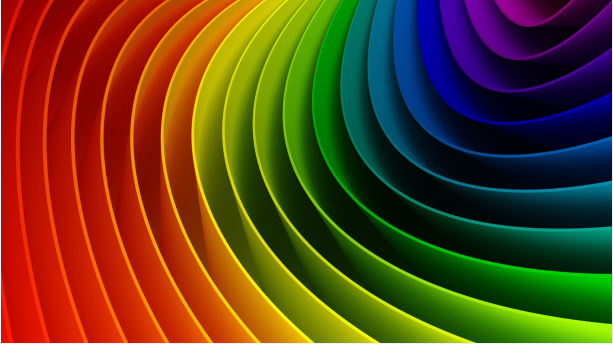
Grayscale főkép futtatás előtt:



Futtatás után 60x60-as bementi paraméterekkel:



Színes főkép futtatás előtt:



Futtatás után 80x80-as bementi paraméterekkel:

