

Concepte și Aplicații în Vederea Artificială

Tema de laborator 2 - Redimensionarea imaginilor cu păstrarea conținutului

Obiectiv

Scopul acestei teme este implementarea și testarea unei versiuni a algoritmului de redimensionare a imaginilor cu păstrarea conținutului (Figura 1) propus de S. Avidan și A. Shamir în articolul "Seam Carving for Content-Aware Image Resizing" (găsiți articolul atașat în materialele pentru laborator). Articolul propune o metodă de a păstra cât mai nealterat conținutul unei imagini în urma redimensionării ei. Conținutul unei imagini este definit de regiunile "interesante" dintr-o imagine, acele regiuni cu gradienti mari.

Funcțiile Python care vă vor ajuta la implementarea temei sunt în directorul *cod*; imaginile pe care le veți folosi sunt în directorul *data*.

Punct de pornire

Scriptul *run_project.py* vă oferă un punct de pornire în implementarea voastră. În acest script se setează imaginea ce urmează a fi redimensionată și sunt setați parametri folosiți. Acest script apelează funcția *resize_image* care la rândul ei apelează funcții pentru micșorarea în lățime sau înălțime a imaginii, amplificarea conținutului sau eliminarea unui obiect.

1.1 Micșorarea imaginii pe lățime - **Pasul 1**

Implementați mai întâi operația de micșorare pe lățime a unei imagini. Funcția *decrease_width* realizează operația de eliminare a pixelilor de pe drumurile verticale ce conectează prima linie cu ultima linie din imagine. Această funcție apelează următoarele funcții:

- *compute_energy* - funcția trebuie completată folosind ecuația (1) din articolul atașat;
- *select_path.py* - scriptul este scris parțial, trebuie completat;
- *show_path* - funcția este scrisă în întregime;
- *delete_path* - funcția este scrisă parțial, trebuie completată.

Drumul optim vertical este drumul de cost minim și se obține prin metoda programării dinamice (vedeți slide-urile de la curs). În script este scris codul pentru o metodă care alege aleator drumuri, fără să țină cont de funcția cost.



Figura 1: **Redimensionare imaginilor.** (a) Imaginea inițială; (b) Imaginea inițială redimensionată la o imagine cu 50 de pixeli mai puțini în lățime folosind algoritmul de păstrare al conținutului; (c) Imaginea inițială redimensionată la o imagine cu 50 de pixeli mai puțini în lățime cu algoritmul de redimensionare uzuală (folosind funcția 'resize' din OpenCV) ce scalează tot conținutul din imagine.

1.2 Micșorarea imaginii pe înălțime - **Pasul 2**

Realizați operația de eliminare a pixelilor de pe drumurile orizontale ce conectează prima coloană cu ultima coloană dintr-o imagine scriind funcția *decrease_height*. Inspirați-vă în scrierea ei din funcția *increase_width*.



Figura 2: **Redimensionare imaginilor.** (a) Imaginea inițială; (b) Imaginea inițială redimensionată la o imagine cu 100 de pixeli mai puțini în înălțime folosind algoritmul de păstrare al conținutului; (c) Imaginea inițială redimensionată la o imagine cu 100 de pixeli mai puțini în înălțime cu algoritmul de redimensionare uzuală (folosind funcția 'resize' din OpenCV) ce scalează tot conținutul din imagine.

Pentru prima parte a temei, implementați pașii 1 și 2 descriși mai sus și realizați următoarele:

- (a) Completați pentru început restul funcției *delete_path* și observați cum rulează scriptul *run_project* pentru cazul drumurilor alese aleator. Apoi completați restul scriptului *select_path.py* pentru cazul 'programareDinamica' și testați codul vostru pe imaginea 'castel.jpg' furnizată eliminând 50, 75 și 100 de pixeli în lățime (Figura 1).
- (b) Testați-vă codul de la pasul 2 pe imaginea 'praga.jpg' furnizată eliminând 50, 75 și 100 de pixeli pe înălțime (Figura 2).
- (c) Rulați algoritmi implementați de voi la pașii 1 și 2 pe imaginile furnizate explicând de fiecare dată de ce algoritmul vostru a reușit sau nu a reușit.

1.3 Amplificarea conținutului imaginilor - Pasul 3

Secțiunea 4.4 din articol descrie cum puteți amplifica conținutul unei imagini păstrând dimensiunile inițiale ale imaginii. Scrieți funcția *amplify_content* care implementează această operație prin scalarea imaginii inițiale cu un anumit factor și apoi eliminarea de drumuri.



(a)



(b)

Figura 3: **Amplificarea conținutului.** (a) Imaginea inițială; (b) Imaginea inițială cu conținutul amplificat folosind un factor de amplificare de 1.2;

1.4 Eliminarea unui obiect din imagine - Pasul 4

Secțiunea 4.6 din articol descrie cum puteți elimina un obiect dintr-o imagine marcat de un utilizator. Scrieți funcția *delete_object* care micșorează imaginea cu scopul de a elimina obiectul delimitat. Folosiți funcția *selectROI(image)* din OpenCV pentru a delimita obiectul.

Pentru a doua parte a temei, implementați pașii 3 și 4 descriși mai sus și realizați următoarele:

- (a) Testați algoritmul de la pasul 3 pe imaginea 'arcTriumf.jpg' furnizată folosind diverși factori la scalarea imaginii inițiale (Figura 3).
- (b) Testați-vă codul de la pasul 4 pe imaginea 'lac.jpg' furnizată (Figura 4).

(c) Rulați algoritmi implementați de voi la pașii 3 și 4 pe diverse imagini de pe Internet explicând de fiecare dată de ce algoritmul vostru a reușit sau nu a reușit.



(a)



(b)

Figura 4: **Eliminarea unui obiect.** (a) Imaginea inițială; (b) Imaginea inițială cu un obiect eliminat