



دانشکده مهندسی کامپیوتر

استاد درس: دکتر رضا انتظاری

۱۳۹۹ پاییز

پروژه پایانی درس طراحی الگوریتم ها

محمد رضا باقری
شماره دانشجویی: ۹۶۵۲۱۰۸۳

۱ مقدمه

SeamCarving یک الگوریتم تغییر ابعاد تصاویر با آگاهی از داده های موجود در تصویر است. این تغییر سائز در جهت افقی یا عمودی یا هردو می تواند باشد. همانطور که می بینید بیشتر جزئیات مهم در تصویر زیر در عین کاهش سائز حفظ شده است.



شکل ۱: ورودی الگوریتم



شکل ۲: خروجی الگوریتم

۱.۱ محاسبه ماتریس انرژی

برای محاسبه ماتریس انرژی، ابتدا یک ماتریس دو بعدی با اندازه تصویر ساخته می شود. سپس با استفاده از فرمول های گفته شده انرژی هر پیکسل محاسبه می شود و هر درایه ماتریس انرژی مقدار دهی می شود.



۲.۱ کد محاسبه ماتریس انرژی

ابتدا یک نمونه از کلاس *SeamCarving* ساخته می شود. سپس با فراخوانی متد `computeEnergyMatrix` ماتریس انرژی محاسبه می شود.

• `computeEnergyMatrix`

• پیچیدگی زمانی این الگوریتم از $O(|M||N|)$ است.

• $O(|M||N|)$ required space

• `computeDeltaX`

• پیچیدگی زمانی این الگوریتم از $O(1)$ است.

• $O(1)$ required space

• `computeDeltaY`

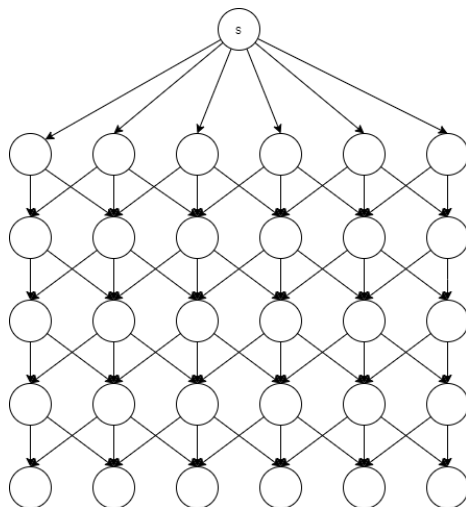
• پیچیدگی زمانی این الگوریتم از $O(1)$ است.

• $O(1)$ required space

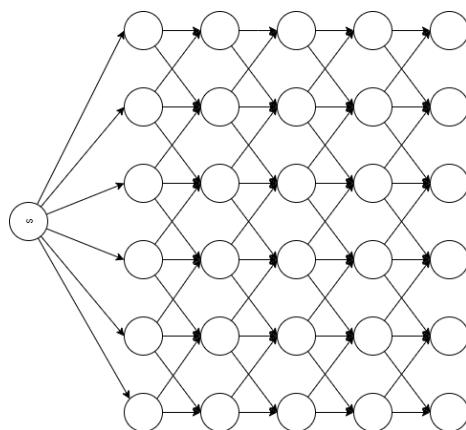
۲ پیدا کردن درز با کمتری انرژی

۱.۲ الگوریتم دایجسترا

در این پروژه برای پیدا کردن یک درز با کمتری انرژی از الگوریتم دایجسترا استفاده می شود. ورودی این الگوریتم ماتریس انرژی است که در مرحله قبلی محاسبه شده است. هر درایه از این ماتریس به منزله یک نود تلقی می شود. و انرژی هر نود بیانگر وزن یک یال بین نود و نود همسایه آن می باشد.



شکل ۳: گراف انرژی برای پیدا کردن یک درز عمودی



شکل ۴: گراف انرژی برای پیدا کردن یک درز افقی

الگوریتم دایجسترا ابتدا **distance table** را محاسبه می کند. با استفاده از این جدول و شروع از نود **S** مسیر با کمترین انرژی محاسبه می شود.

راس **S** را در نظر می گیریم. برای پیدا کردن یک درز عمودی، راس **S** یک راسی است که با تمام راس های ردیف اول همسایه است. همین راس هنگام پیدا کردن یک درز افقی، یک راسی است که با تمام راس های ستون اول ماتریس انرژی همسایه است.

پس از محاسبه جدول **distance table**، از پایین ترین ردیف ماتریس انرژی و با توجه به جدول **distance table** یک راس با کمترین انرژی محاسبه شده از راس **S** را پیدا می کنیم. با توجه به اینکه ما در این الگوریتم **Parent** هر راس را نگه داشته ایم، می دانیم با انتخاب یکی از رئوسی که در پایین ترین ردیف در ماتریس انرژی قرار دارند، می توان به راس **S** رسید.



- `computeDistance`

- $O(|E| + |V|\log|V|)$

- $O(n^2)$ required space

-

- `getPath`

- $O(|M| + |N|)$

- $O(|M| + |N|)$ required space

-

- `getCost`

- $O(1)$

- $O(1)$ required space

-

- `getSuccessor`

- $O(|M| + |N|)$

- $O(|M| + |N|)$ required space

-

- `getSeam`

- $O(|M| + |N|)$

- $O(1)$ required space

۳ رسم و حذف درز پیدا شده

۱.۳ رسم درز

پس از پیدا کردن یک درز، نوبت به رسم درز در تصویر انرژی می شود. متد های زیر برای رسم یک درز بر روی تصویر فراخوانی می شوند.

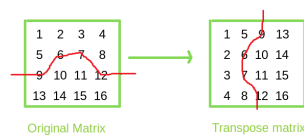
- `plotSeam`

- $O(|E| + |V|)$

- $O(1)$ required space

۲.۳ حذف درز

حذف درز از تصویر هم بسته به عمودی یا افقی بودن درز متفاوت است. در صورتی که یک درز عمودی داشته باشیم، به منظور حذف آن نود های مشترک در تصویر و مسیر حذف می شود. به منظور حذف یک درز افقی، ابتدا ترانواده ماتریس عکس محاسبه می شود و سپس نود های مشترک بین تصویر و درز حذف می شوند. سپس از ماتریس حاصل مجددا ترانواده گرفته می شود. لازم به ذکر است که درز محاسبه شده هم از ماتریس عکس و هم از ماتریس انرژی حذف می شود. دلیل حذف درز از ماتریس انرژی این است که مجددا مجبور به محاسبه ماتریس انرژی نباشیم.



شکل ۵: حذف یک درز افقی از عکس

`removeVerticalSeamFromEnergy` •

$O(|M| + |N|)$ •

$O(1)$ required space •

`removeHorizontalSeamFromEnergy` •

$O(|N||M|)$ •

$O(|M||N|)$ required space •

`removeVerticalSeam` •

$O(|N| + |M|)$ •

$O(1)$ required space •

`removeHorizontalSeam` •

$O(|M||N|)$ •

$O(|M||N|)$ required space •



پروژه پایانی درس طراحی الگوریتم ها

- `createResizedImage`

- $O(|M||N|)$

- $O(|M||N|)$ required space

۴ اجرای پروژه

به منظور اجرای پروژه، می بایست دستور زیر اجرا شود. در انتهای دستور تعداد درز های افقی و عمودی که می بایست پیدا شوند مشخص می شود.

```
>>> python main.py -i <inputfile> -h <horizontalSeam> -v <verticalSeam>
# -h = Number of horizontal seam to find
# -v = Number of vertical seam to find
>>> python main.py -i test_case_1.png -h 10 -v 10
```

Listing : \ External file code