

# به نام خدا

پروژه حذف لرزش‌های تصویر

درس تجزیه و تحلیل سیگنال‌ها و سیستم‌ها  
دکتر محمدی

حامد کلانتری سرچشمه

شماره دانشجویی: ۹۴۵۲۱۱۹۸

سید رامتین موسوی گرایش

شماره دانشجویی: ۹۵۵۲۱۴۵۹

(اسامی براساس حروف الفبا می‌باشند)

نیمسال اول سال تحصیلی ۹۷-۹۸

ما در اولین تلاش برای انجام این پروژه از داده‌های ارائه شده نمونه که تغییرات هرفریم را بدست آورده بود استفاده کردیم (فقط تغییرات طولی و عرضی را شامل میشد) تا به این ترتیب مرحله اول الگوریتم کنونی پیاده‌سازی شده را اجرا نکنیم اما باتوجه به خطاهای خروجی که تصویر را بدون هیچ الگوی خاصی جابجا میکرد و عملاً اشکال زدایی و دیباگ آن را غیرممکن بود تصمیم برآن شد تا مرحله اول را نیز پیاده سازی کرده و تغییرات فریم را به دست آوریم (علاوه بر بدست آوردن تغییرات طولی و عرضی که برای لرزش‌های بالا-پایین و چپ-راست کاربرد دارند بخشی تحت عنوان تغییرات زاویه را هم اضافه کردیم تا تغییرات چرخشی و زاویه‌ای هرزاویه را نیز پوشش دهد) که نتیجه بسیار مطلوب تری را شاهد بودیم.

روش به کار رفته در این پروژه از دو ساختار استفاده میکند:

۱- پارامترهای حرکت

۲- خط سیر تصویر

قسمت اول اطلاعات مربوط به تغییرات هر فریم با فریم های قبلی را نگه میدارد که شامل تغییرات در راستای محور طول ها ، محور عرض ها و تغییرات مربوط به زاویه هر فریم با فریم قبلی می باشد (dx, sy, da).

قسمت دوم و ساختار دومی که از آن استفاده میکنیم در واقع همین پارامترهای حرکت و تغییرات زاویه‌ای را نسبت به یک نقطه خاص بررسی میکنیم (در اینجا نسبت به اولین فریم تصویر) که یک مسیر مشخص از خط سیر دوربین به ما میدهد.

## مرحله اول

برای قسمت اول و یافتن تغییرات میتوانیم یک محدوده خوب (منظور از محدوده خوب، نقطه‌ای است که انرژی آن زیاد باشد یا درواقع تراکم پیکسلی زیادی داشته باشد) را انتخاب کنیم و در دو فریم پیاپی آن محدوده را به کمک کتابخانه‌های مختلف و الگوریتم‌های متفاوت شناسایی کرده و تغییرات آن نسبت به فریم قبلی را بدست آوریم.

## مرحله دوم

در قسمت دوم نیز برای بدست آوردن خط سیر تصویر میتوانیم خط سیر هر فریم از تصویر را نسبت به فریم اولیه (همان نقطه خاص) بدست آوریم به این روش که که مجموع جبری فریم‌های قبلی تا فریم مورد نظر را حساب کنیم که این درواقع همان تغییرات فریم  $n$  ام نسبت به اولین فریم می‌باشد. (این کارکرد را تابع accumulator در این پروژه انجام میدهد و ورودی آن مجموعه نقاط dx, dy, da می‌باشد)

## مرحله سوم

در این مرحله ما تمامی داده های لازم برای پردازش هر فریم از ویدیو را داریم و اکنون در این مرحله لرزش ویدیو را از بین میبریم. در این مرحله به ازای خطسیر هر فریم از تصویر میانگین تغییرات  $r$  فریم بعدی و  $r$  فریم قبلی را به دست می‌آوریم، که درواقع به  $r$  شعاع هموار سازی می‌گوییم. اصطلاحاً این روش لرزش گیری از تصویر را sliding average window algorithm می‌گویند.

$$Sum = \sum_{i=k-r}^{i=k+r} I_{original}^i$$

$$WindowSize = 2 * r$$

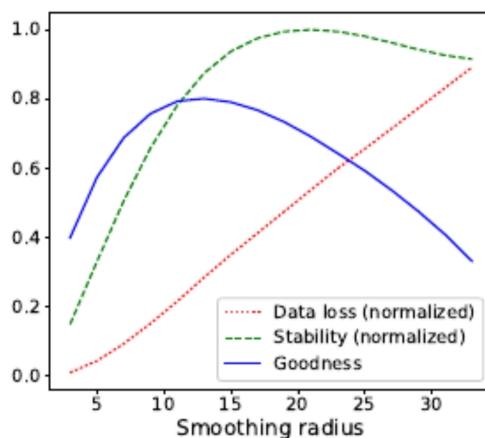
$$I_{smooth}^k = Sum / WindowSize$$

$I(\text{original})$  همان خطسیر فریم می‌باشد که در مرحله دوم به دست آوردیم و  $I(\text{smooth})$  همان میانگین تغییرات می‌باشد.

همانگونه که مشاهده می‌شود بزرگترین چالش ما برای بدست آوردن تصویر بدون لرزش استفاده از بهینه ترین  $r$  (شعاع هموار سازی) می‌باشد. در صورتی که شعاع را خیلی کوچک در نظر بگیریم درواقع تغییری در موقعیت فریم‌ها ایجاد نمیشود و درواقع خروجی ما لرزش خواهد داشت و در صورتی که شعاع خیلی بزرگ در نظر گرفته شود تصویر ما دچار  $\text{loss data}$  میشود که دلیل این مهم نیز زیاد شدن تعداد فریم‌ها میباشد که در نتیجه باعث میشود اشتراک تعداد زیادی از فریم‌ها محدوده کمی از تصویر را پوشش دهد و بخشی از صفحه را از دست بدهیم اما همین افزایش شعاع باعث ثبات تصویر و عدم لرزش آن می‌شود. حال در صورتیکه شاخصی را به عنوان معیاری از خوب بودن خروجی بصورت زیر تعریف کنیم، که در آن  $s(r)$  ثبات تصویر (stability) و  $d(r)$  از بین رفتن اطلاعات (data loss) فرض شوند، برای داشتن بهترین خروجی باید مقدار تابع  $G$  بیشینه شود:

$$G(s_V(r), d_V(r)) = \sqrt{s_V(r) * (1 - d_V(r))}$$

که باتوجه به آزمایشات و تلاش های صورت گرفته برای تعداد زیادی از داده‌ها نتایج زیر بدست آمده:



که میتوان درمورد رابطه افزایش شعاع و تغییر کیفیت تصاویر به این داده‌ها استناد کرد و بهترین شعاع را انتخاب نمود. عموماً برای بهترین نتیجه شعاع را برابر ۴۰ در نظر میگیرند. پیاده‌سازی این قسمت از پروژه نیز در تابع `low_filter` انجام گرفته است که ورودی آن نتایج به‌دست آمده از مرحله دوم می‌باشد.

## مرحله چهارم

حال با داشتن تمامی این اطلاعات در مرحله بعدی باید میزان جابجایی‌ای که هر فریم لازم دارد تا با اعمال آن لرزش فریم مربوطه حذف شود را به دست می‌آوریم و سپس در مرحله بعدی روی تک به تک فریم‌های تصویر اعمال میکنیم.

برای به دست آوردن این میزان جابجایی مقدار  $I(\text{smooth})$  را از  $I(\text{original})$  کم کرده و با نتایجی که در مرحله اول به دست آوردیم بصورت تک به تک روی فریم‌ها اعمال می‌کنیم. درواقع یک نقطه از تصویر قبلی به نقطه کنونی در فریم جدید نگاشت

می‌شود و از آنجا به نقطه میانگین بدست آمده در مرحله سوم نگاشت می‌گردد. در واقع تبدیلی که روی این مجموعه از داده‌ها برای ازبین بردن لرزش تصویر اعمال می‌کنیم به صورت زیر می‌باشد که روی تک به تک فریم‌های ویدیو اعمال می‌شود:

$$T_{smooth}^{(k)} = \begin{bmatrix} \cos(da^{(k)}) & -\sin(da^{(k)}) & dx^{(k)} \\ \sin(da^{(k)}) & \cos(da^{(k)}) & dy^{(k)} \end{bmatrix}$$

این مرحله از پروژه نیز توسط تابع prev\_to\_current در برنامه انجام می‌گردد که دو ورودی دارد به این صورت که ورودی اول داده‌های بدست آمده از مرحله سوم و ورودی دوم آن داده‌های بدست آمده از مرحله دوم می‌باشد.

### مرحله پنجم

در این مرحله صرفاً داده‌های بدست آمده از مرحله چهارم را روی تک به تک فریم‌های تصویر اعمال کرده و نتایج را نمایش می‌دهیم.

الگوریتم بکار رفته در این پروژه به نحو زیر می‌باشد:

---

#### Algorithm 1: Stabilisation using sliding window

---

**Input:** Shaky camera footage

**Parameter :** Smoothing radius,  $r$

**Output:** Stabilized video

- 1 Process each frame to obtain frame-to-frame transform  $\Delta_{original}$
  - 2 Using  $\Delta_{original}$ , compute trajectories  $I_{original}$
  - 3 **while**  $k < \text{Number of frames} - r$  **do**
  - 4      $Sum = \sum_{i=k-r}^{i=k+r} I_{original}^i$
  - 5      $WindowSize = 2 * r$
  - 6      $I_{smooth}^k = Sum / WindowSize$
  - 7 **end**
  - 8 Compute  $\Delta_{smooth}$  to raise each point from its position in  $I_{original}$  to  $I_{smooth}$
  - 9 Using  $\Delta_{smooth}$ , obtain and apply transformation matrices  $\Gamma$  on the sequence of frames
  - 10 Resultant sequence of frames is the stabilized output
- 

منابع:

[1] Keerthan S. Shagrithaya etc., Video Stabilization Using Sliding Frame Window, 2017