پروژه پایانترم

دیبا امین شهیدی

|  |  |  |
| --- | --- | --- |
| اطلاعات گزارش |  | چکیده |
| **تاریخ: ۱۲/۱۱/۱۳۹۸** |  | در اين گزارش‏‌ ، مراحل اصلی یک روش **Image Registration** روی . چندین نوع نمونه عکس که داری **noise** های مختلف و دارای دوران و تغیر سایز های مختلف هستند مورد بحث قرار گرفته است. اقدامات مختلف عملکرد ارائه می شود که کیفیت ثبت و دقت ثبت را تعیین می کند.. |
| **واژگان كليدي:**  **Image Registration** |  |

1-مقدمه[[1]](#footnote-1)

ImageRegistration فرایندی است که تصاویری در زمان های مختلف و در چند نقطه دید از حسگرهای مختلف ضبط می شوند می توان مورد تجزیه و تحلیل قرار داد . در این فرایند ابتدا نقاط خاص هر عکس شناسایی شده (فارق از زاویه عکس نور و نویز های رو عکس ها ) و برای تطبیق این نقاط روی هم قرار میگیرند ، به این صورت میتوانیم دو یا چند تصویر را روی هم تطبیق دهیم . این فرایند در علوم پزشکی ، سنجش از دور و بینایی کامپیوتر کاربرد دارد.

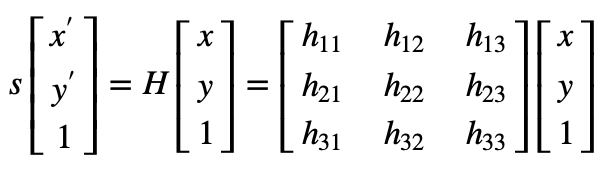
2-شرح تکنیکال

برای انجام این فرایند ۳ مجموعه داده داریم خروجی هر مرحله ورودی مرحله بعد است و ۱ مجموعه نهایی که باید خروجه فرایند را با عکس اصلی مقایسه کرد .

در مجموعه داده اول ( Attack 1 ) دارای تغیرات هندسی و فیلتر شده مجموعه ( Halftone) است و مجموعه داده (Halftone) که به عنوان مجموعه مرجع اولیه استفاده مجموعه میشود درواقع نویز گرفته عکس های اصلی هستند . و در مجموعه داده (Attack 2) درواقع همان تغیرات هندسی که در ( Attack 1 ) اعمال شده است به روی عکس اصلی میباشد . بنابر برای انجام این فرایند میتوان با استفاده از دو عکس( Attack 1 )  و ( Halftone) متناظر نقاط حساس اصلی مشترک را پیدا کرد و با تطبیق ان ها تغیرات هندسی اعمال شده رو ان عکس را پیدا کرد و با اعمال ان ها روی عکس(Attack 2) متناظر به عکس اصلی دست یافت . روش اعمال این فرایند ها به ترتیب توضیح داده می شود .

ماتریس هوموگرافی : در بینایی کامپیوتر ، هر دو تصویر با فضا یکسان در فضا توسط یک هموگرافی مرتبط هستند.

به طور خلاصه ، هموگرافی مسطح مربوط به تغییر و تحول بین دو صفحه (تا یک عامل مقیاس) است .



برای مثال در واقعیت افزوده ، حذف / اصلاح دیدگاه و Panorama stitching کاربر فراوان دارد .

روش : OBR

این روش یک روش جایگزین کارآمد برای SIFT یا SURF می باشند که از ترکیب دو روش FAST keypoint detector و Brife است .

* پیدا کردن Feature برای تشخیص نقاط مشابه به روش ORB
* اعمال این ویژگی ها روی عکس مجموعه داده اول و Halftone متناظر ان به روش Brute force
* تطبق ویژگی ها مشترک برای پیدا کردن ماتریس homography

2-شرح نتایج

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # | Type 1 (Histeq) | | | Type 2 (Sharpen) | | | Type 3 (Gaussfilt) | | | Type 4 (Noise) | | | Type 5 (Bilatfilt) | | | Mean | | | STD | | |
| **SSIM** | **MSE** | **MP** | **SSIM** | **MSE** | **MP** | **SSIM** | **MSE** | **MP** | **SSIM** | **MSE** | **MP** | **SSIM** | **MSE** | **MP** | **SSIM** | **MSE** | **MP** | **SSIM** | **MSE** | **MP** |
| Sample 1 | 0.675711559 | 445.1682587 | 1794 | 0.647317123 | 421.8084488 | 1593 | 0.237960125 | 8358.387653 | 149 | 0.856116584 | 1090.008343 | 1666 | 0.812544495 | 300.9891434 | 1231 | 0.645929977 | 2123.272369 | 1286.6 | 0.218754234 | 3129.729259 | 598.7495637 |
| Sample 2 | 0.832868034 | 161.53339 | 1655 | 0.829202045 | 169.9427605 | 1317 | 0.336213099 | 6037.220207 | 139 | 0.68288394 | 1214.45541 | 1359 | 0.522378678 | 1293.273434 | 1033 | 0.640709159 | 1775.28504 | 1100.6 | 0.190238412 | 2185.967733 | 519.6543467 |
| Sample 3 | 0.894030116 | 90.09452438 | 2213 | 0.902183187 | 41.87719727 | 1642 | 0.494372472 | 7815.896149 | 347 | 0.741564719 | 1502.448837 | 1655 | 0.850797635 | 75.2492218 | 1126 | 0.776589626 | 1905.113186 | 1396.6 | 0.152280075 | 3007.119453 | 627.4368813 |
| Sample 4 | 0.884240896 | 53.01252365 | 1563 | 0.862351888 | 107.5873985 | 1160 | 0.498048565 | 7550.14344 | 268 | 0.856116439 | 1267.270576 | 1226 | 0.902702551 | 66.45746994 | 764 | 0.800692068 | 1808.894282 | 996.2 | 0.152219203 | 2907.540199 | 443.867728 |
| Sample 5 | 0.943274317 | 37.71850204 | 1535 | 0.861164562 | 98.26182938 | 1576 | 0.516468457 | 9752.476219 | 237 | 0.825002702 | 1713.047352 | 1531 | 0.827734666 | 151.3036079 | 1061 | 0.794728941 | 2350.561502 | 1188 | 0.145543255 | 3753.760717 | 511.6897498 |
| Sample 6 | 0.89908933 | 66.0461235 | 2629 | 0.940080267 | 37.42256165 | 2139 | 0.531904811 | 7308.269855 | 814 | 0.877258175 | 1103.041363 | 2512 | 0.950231732 | 40.26744461 | 1432 | 0.839712863 | 1711.00947 | 1905.2 | 0.156185177 | 2828.325494 | 687.2249704 |
| Sample 7 | 0.886135963 | 313.2577896 | 1731 | 0.851741588 | 745.8494377 | 2045 | 0.524054182 | 16190.64301 | 829 | 0.891022581 | 2682.076565 | 2377 | 0.943956512 | 103.2182503 | 1533 | 0.819382165 | 4007.00901 | 1703 | 0.150580833 | 6159.787747 | 522.4557398 |
| Sample 8 | 0.876476298 | 253.6793251 | 2008 | 0.921894568 | 104.4104195 | 1868 | 0.499932469 | 13918.40275 | 706 | 0.886053733 | 2161.229851 | 2112 | 0.860171218 | 165.9829025 | 1373 | 0.808905657 | 3320.741049 | 1613.4 | 0.155805456 | 5354.60641 | 519.6728201 |
| Sample 9 | 0.732599005 | 189.0021706 | 1517 | 0.802952307 | 56.05301285 | 1189 | 0.363572498 | 9478.74374 | 227 | 0.70752878 | 1389.561897 | 1209 | 0.403134921 | 3131.118229 | 582 | 0.601957502 | 2848.89581 | 944.8 | 0.181642086 | 3493.971769 | 469.8316294 |
| Sample 10 | 0.824907283 | 150.8796844 | 1856 | 0.806821416 | 179.9765663 | 1511 | 0.336109554 | 8074.632164 | 289 | 0.653052329 | 1340.074345 | 1424 | 0.668460371 | 441.1953468 | 1243 | 0.657870191 | 2037.351621 | 1264.6 | 0.175358563 | 3049.304356 | 526.9036345 |
| Mean | 0.84493328 | 176.0392292 | 1850.1 | 0.842570895 | 196.3189632 | 1604 | 0.433863623 | 9448.481518 | 400.5 | 0.797659998 | 1546.321454 | 1707.1 | 0.774211278 | 576.905505 | 1137.8 | 0.738647815 | 2388.813334 | 1339.9 | 0.16786073 | 3587.011314 | 542.7487064 |
| STD | 0.078136828 | 123.3093596 | 333.3788986 | 0.078310588 | 211.778791 | 318.2970311 | 0.09956234 | 3018.149511 | 258.6976034 | 0.087064966 | 485.5521729 | 444.6199388 | 0.17498457 | 922.296229 | 279.9052697 | 0.085732345 | 728.0336478 | 298.7483255 | 0.022142221 | 1166.822661 | 69.97806615 |

**مراجع**

https://www.geeksforgeeks.org/image-registration-using-opencv-python/

https://opencv-python-tutroals.readthedocs.io/en/latest/py\_tutorials/py\_feature2d/py\_orb/py\_orb.html

https://en.wikipedia.org/wiki/Homography\_(computer\_vision)

1. [↑](#footnote-ref-1)