فشرده سازی بدون اتلاف برای تصاویر Stereoscopic

آرین تشکر

چكيده

در اين مقاله، روشی برای فشرده سازی بدون اتلاف تصاویر Stereoscopic ارائه می­شود. با توجه به این که تصاویر Stereo از یک سوژه و صرفاً با استفاده از دو (یا چند) دوربین اخذ شده اند، میزان وابستگی بین آن ها بسیار زیاد است و می­توان از این وابستگی برای فشرده سازی یکی از تصاویر استفاده است. در این مقاله از یکی از تصاویر به عنوان تصویر Reference استفاده می­کنیم و آن را با روش پیشگویی AdaGAP فشرده سازی خواهیم کرد. سپس تصویر دیگر را با استفاده از تصویر اول به وسیله­ی تصویر و به کمک یک الگوریتم تطابق بلوکی بازسازی میکنیم و خطای بازسازی را به همراه بردار Disparity برای Decoder ارسال خواهیم کرد. در نتیجه­ی این پژوهش 35.6% بهبود در میزان متوسط BPP دو عکس مشاهده می­شود.

پیاده سازی این روش به زبان MATLAB به همراه این مقاله ضمیمه شده است.

كلمات كليدي

فشرده سازی بدون اتلاف، Stereo Matching، RLE، فشرده سازی با استفاده از پیشگو ها (Predictors)

# مقدمه

امروزه دوربین های Stereoscopic برای مصارف مختلفی مورد استفاده قرار می­گیرند که از جمله­ی آنها می­توان به تشخیص عمق و تولید تصاویر 3 بعدی اشاره کرد که موارد پر کاربردی هستند که با توجه به استفاده از چندین دوربین بدیهتاً پهنای باند چند برابری نسبت به یک دوربین معمولی استفاده می­کنند. با این حال، از آنجایی که تصاویر استریو عموماً از یک سوژه گرفته می­شوند، وابستگی بسیار زیادی نسبت به هم دارند، امری که می­توان از آن برای فشرده سازی اطلاعات موجود در تصاویر استریو از آن استفاده کرد و به جای ذخیره سازی تمامی­ تصاویر به صورت فشرده نشده، یک تصویر را به عنوان مرجع (Reference) انتخاب کرده، و اختلاف باقی تصاویر را بر اساس معیاری، با تصویر مرجع به خود آن تصاویر، ذخیره کنیم. در صورتی که بتوانیم باقی تصاویر را طوری نسبت به تصویر مرجع پیشگویی کنیم که خطای پیشگویی کم باشد، آنگاه انتظار می­رود بتوان آنتروپی تصویر پیشگویی شده را تا حد قابل ملاحظه ای کاهش داد.

در این مقاله برای انجام پیشگویی از تطابق بلوکی استفاده می­کنیم و با استفاده از تصویر سمت چپ به عنوان مرجع، سعی می­کنیم که تصویر سمت راست را با حداقل خطا بازسازی کنیم.

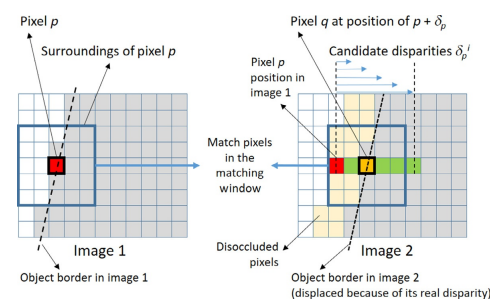
علاوه بر آن، برای فشرده سازی تصویر مرجع از پیشگوی AdaGAP استفاده می­کنیم. تمام عملیات انجام شده کاملاً برگشت پذیر و بدون اتلاف است.

# فرضیات و تعاریف اولیه

تصویر استریو عنوانیست که به تصاویری اطلاق می­شود که از یک سوژه در یک زمان و از زوایای مختلف گرفته شده اند. ویژگی مهم این تصاویر آن است که از آن ها می­توان برای عمق اشیا و سوژه های موجود در محیط را تخمین زد (مشابه کاری که چشم انسان برای تخمین عمق محیط انجام می­دهد).

تطبیق استریو (Stereo Matching) به عمل تخمین مکان یک پیکسل در واقعیت، بر روی هر یک از تصاویر استریو گفته می­شود[4] .

در فرآیند تطبیق استریو فرض می­شود که تصاویر در یک صفحه­ی قرار گرفته اند (Coplanar). در این مقاله نیز بدون انجام Rectification روی جفت تصویر های استریو، فرض شده است که تصاویر از دو دیدگاه در یک صفحه گرفته شده اند. و عملیات تطبیق استریو مستقیماً روی تصاویر صورت پذیرفته است.



شکل 1: تطابق استریو[4]

# طرح مسئله

مسئله­ی مطرح در این مقاله، فشرده سازی تصاویر استریو به صورت بودن اتلاف است. از آنجایی که ذخیره سازی تصاویر و استریم های استریو بدون وجود یک مکانیزم فشرده سازی مستلزم فضای چند برابری نسبت به تصاویر یک دوربین معمولی هستند، باید راهکاری برای ذخیره سازی این تصاویر به صورت بدون اتلاف اندیشید. لازم به ذکر است که مکانیزم ارائه شده در ادامه­ی این مقاله را می­توان به مصارف کلان تر – مثلاً برای فشرده سازی چندین تصویر گرفته شده از یک سوژه با دوربین های مختلف و در فواصل زمانی متفاوت - نیز بسط داد که انگیزه ای قوی تر برای ادامه کار ایجاد می­کند.

# راه حل پیشنهادی

برای فشرده سازی بدون اتلاف جفت تصویر های استریو،روش پیشنهادی ما استفاده از وابستگی مکانی بین این تصویر استفاده می­کند. از آنجایی که جفت تصویر ها ویژگی های بصری بسیار مشابهی با یکدیگر دارند، می­توان انتظار داشت که بتوان با تطابق این ویژگی ها، یکی از عکس ها را با خطای کم طوری بازسازی کرد که فقط از تصویر مرجع استفاده شده باشد.

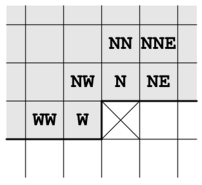
به علاوه، برای فشرده سازی تصویر مرجع که از یک روش فشرده سازی مبتنی بر پیشگویی به نام AdaGAP استفاده می­شود.

## AdaGAP

پیشگوی AdaGAP در حقیقت ترکیبی از روش های GAP و ALCM برای پیشگویی Adaptive پیکسل ها در Raster Scan می­باشند.

در روش GAP از 7 همسایه­ی علی یک پیکسل برای پیشگویی مقدار آن پیکسل استفاده می­شود. این همسایه های علی را می­توانید در شکل (2) مشاهده کنید. در این روش، با استفاده از محاسبات انجام شده بر روی این همسایه های علی، وضعیت پیکسل فعلی در یکی از چهار دسته­ی اصلی (لبه عمودی قوی یا ضعیف و لبه­ی افقی قوی یا ضعیف) قرار می­گیرد و بر اساس آن مقدار پیکسل فعلی تخمین زده شده و خطای تخمین با مقدار واقعی محاسبه می­شود.

در پیشگوی AdaGAP علاوه بر استفاده از روش GAP، از تکنیکی مشابه به ضرایب Adaptive در روش ALCM استفاده می­شود. از آنجایی که سهم غالب پیشگویی در روش GAP را پیکسل های سمت چپ و بالای پیکسل فعلی تشکیل می­دهند، در ابتدای کار به هر دوی این پیکسل ها ضریب 1 را اختصاص می­دهیم. سپس در صورتی که مقدار خطای تخمین از یک حد آستانه که به صورت تجربی تعیین می­شود، کوچکتر بود، بسته به مثبت و یا منفی بودن خطا مقدار ضریب پیکسل غالب استفاده شده برای پیشگویی پیکسل فعلی را به اندازه ای مشخص تغییر می­دهیم. نتیجه­ی این کار ترکیب دقت پیشگوی GAP [1]در تشخیص لبه ها به همراه عملکرد بهتر در نواحی هموار تصویر است.



شکل 2: 7 همسایه علی استفاده شده در روش GAP [1]

## تطابق استریو بلوکی

برای پیشگویی تصویر دوم از Block Stereo Matching [3]استفاده می­شود. در این روش با توجه به این فرض که تصاویر هم صفحه هستند، می­توان گفت که مکان هر پیکسل در یکی از تصاویر با یک جابجایی در راستای افقی باید بر مکان همان پیکسل در تصویر دیگر منطبق شود[2] . در این مورد استثناهایی نیز وجود دارد که از جمله­ی آنها می­توان به موارد زیر اشاره کرد:

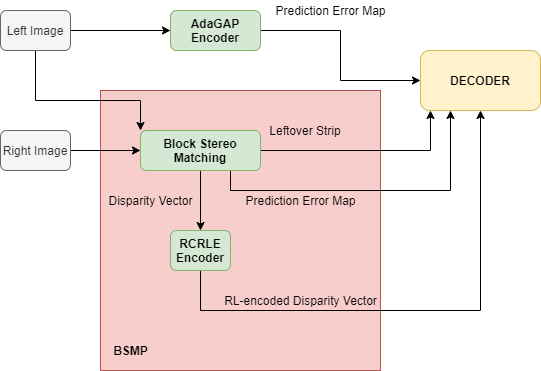
* ­نقاطی از دو تصویر که به واسطه­ی محدودیت میدان دید دوربین ها در تصویر دیگر موجود نیست – مثلاً نظیر سمت جپ ترین نوار های عمودی تصویر سمت چپ را نمی­توان در سمت چپ ترین نوار های تصویر راست یافت و برعکس.
* نقاطی از تصویر که به دلیل پوشیده شدن آن تصویر یک شی دیگر در یکی از دوربین ها قابل مشاهده و در دیگری نظیری ندارند.

با وجود این موارد، تطابق استریو روش مطلوبی برای پیشگویی تصویر دوم بر اساس تصویر مرجع ارائه می­کند که ترکیب آن با AdaGAP برای تصویر مرجع حدود آنتروپی خروجی نهایی را حدود 35.6 درصد کاهش می­دهد. تطابق استریو بلوکی در این روش شامل دو ماژول Encoder و Decoder است که در ادامه به بررسی هر کدام خواهیم پرداخت.

### ماژول Encoder

این ماژول تصویر با دریافت تصاویر سمت چپ و راست، عملیات تطابق استریوی بلوکی را بر روی تصویر راست با استفاده از تصویر چپ به عنوان مرجع انجام می­دهد. برای این کار مراحل زیر به ترتیب طی می­شوند:

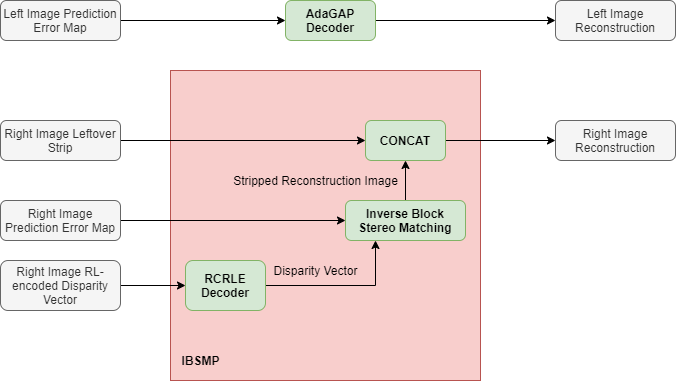
1. ابتدا تصویر به یک شبکه­ی مربعی به ضلع block\_size تقسیم بندی می­شود. در این تقسیم بندی در صورتی که عرض تصویر (width) به block\_size قابل قسمت نباشد، قسمت باقی مانده آن جدا شده و به عنوان Strip بازگردانده خواهد شد. اگر طول تصویر به block\_size قابل قسمت باشد ایرادی ندارد. علت این امر این است که عملیات تطابق فقط در راستای طول تصویر انجام می­شود و بنابراین برای بلوک های سمت راست تصویر که با باقی بلوک ها برابر نیستند نمی­توان تطبیقی پیدا کرد. بدیهی است که طول Strip بین 0 تا block\_size-1 متغیر است.
2. سپس به ازای هر بلوک از تصویر سمت راست، از بلوک متناظر در تصویر سمت چپ و تا window\_size تا بلوک هم پوشان بعد از آن در جهت طول تصویر، با بلوک فعلی انتخابی از تصویر سمت راست تطابق داده می­شوند و بهترین بلوک بر اساس معیار SAD انتخاب می­شود. دلیل استفاده از SAD به عنوان تابع هزینه سرعت فوق العاده بالای آن نسبت به تابعی مانند MSE است که به ما اجازه می­دهد window\_size های بزرگتری را در زمان کمتر بررسی کنیم[2, 5] .
3. میزان جابجایی بهترین بلوک تطبیق داده شده در مرحله­ی قبل با بلوک فعلی از تصویر سمت راست در یک ماتریس Disparity قرار می­گیرد که نشان می­دهد هر پیکسل از تصویر سمت چپ را باید چقدر به سمت راست ببریم تا به بازسازی ممکن از تصویر سمت راست برسیم. بعد از بازسازی عکس در Encoder، میزان خطای آن به صورت خروجی بهDecoder ارسال می­شود.
4. وجود تعداد زیادی عنصر برابر هم در ماتریس Disparity، آن را به یک هدف مناسب برای RLE تبدیل می­کند. بنابراین بعد از استخراج ماتریس Disparity آن را به یک انکودر RLE می­دهیم که به صورت طولی و عرضی ماتریس را کد می­کند و خروجی آن آنتروپی RLE بهینه و ماتریس Disparity بازسازی شده است. (ماژول RCRLE در شکل (3))



شکل 3: ماژول Encoder

### ماژول Decoder

در ماژول Decoder عکس عملیات Encoder انجام می­گیرد. با دریافت خطای پیشگویی و ماتریس Disparity، ماژول Decoder می­تواند تصویر سمت راست را با شیفت دادن هر پیکسل از تصویر سمت چپ به سمت راست به اندازه­ی عدد متناظر همان پیکسل در ماتریس Disparity، و سپس جمع کردن تصویر حاصل با ماتریس خطا، بازسازی کند. در نهایت قطعه­ی باقی مانده عکس که تحت Strip از آن جدا شده بود به سمت راست تصویر چسبانده می­شود و تصویر نهایی به عنوان خروجی این ماژول برگردانده خواهد شد.



شکل 4: ماژول Decoder

# آزمایش ها و نتایج

روش پیشنهادی بر روی 5 جفت تصویر استریو مورد آزمایش قرار گرفت و در نهایت متوسط BPP جفت تصاویر با متوسط آنتروپی تصاویر اولیه به عنوان معیار مقایسه استفاده شد. توجه کنید که با توجه به RGB بودن تصاویر، مقصود از BPP در اینجا تعداد بیت مصرفی به ازای هر پیکسل در هر صفحه­ی رنگی از تصویر است. همچنین آنتروپی تصاویر با استفاده از فرمول (1) بدست آمده است،

(1)

که احتمال وقوع سمبل s در جامعه آماری مورد بررسی است. همچنین برای محاسبه­ی آنتروپی کل چند جامعه آماری از میانگین وزن دار آن ها به نسبت تعداد سمبل هایشان استفاده شده است. در جدول (1) میزان بهبود آنتروپی تصاویر بعد از اعمال راه پیشنهادی را می­توان مشاهده کرد. همانطور که انتظار می­رود به دلیل رخ دادن پدیده Occlusion برای تصاویری که ساختار پیچیده تری دارند و نواحی پرفرکانس آن ها بیشتر است بهبود کمتری حاصل شده است. همچنین در جدول (2) زمان مصرفی بخش های مختلف الگوریتم به ازای هر تصویر نشان داده شده است. همانطور که مشخص است بیشتر زمان مصرف شده مربوط به پیشگوی AdaGAP است که بهبود زمان اجرای یکی از اهداف اصلی در پژوهش های آینده­ی این مقاله است.

|  |  |  |  |
| --- | --- | --- | --- |
| درصد بهبود | میانگین آنتروپی بعد از فشرده سازی | میانگین آنتروپی بدون فشرده سازی | نام جفت تصویر استریو |
| 31.43 | 5.1241 | 7.4724 | Aloe vera |
| 37.96 | 4.5806 | 7.3829 | Books |
| 35.71 | 4.9601 | 7.7148 | Dolls |
| 35.05 | 5.0037 | 7.7045 | Snowman |
| 37.97 | 4.8359 | 7.7959 | Teddy |
| 35.62 | 4.9009 | 7.6141 | Average |

جدول 1: میزان بهبود جفت تصویر های مورد آزمایش

|  |  |  |  |
| --- | --- | --- | --- |
| زمان کل(s) | میانگین زمان تطبیق و عکس تطبیق بلوکی استریو(s) | میانگین زمان enc و dec ماژول (s)AdaGAP | نام جفت تصویر استریو |
| 22.8 | 0.33 | 11.7 | Aloe vera |
| 18.1 | 0.35 | 8.7 | Books |
| 18.9 | 0.35 | 9.1 | Dolls |
| 21 | 0.34 | 10.1 | Snowman |
| 20.2 | 0.35 | 9.78 | Teddy |
| 20.2 | 0.35 | 9.76 | Average |

جدول 2: زمان اجرای الگوریتم

# نتيجه گیری

در این مقاله روشی برای فشرده سازی بدون اتلاف تصاویر استریو با استفاده از وابستگی مکانی بین جفت تصاویر ارائه شد که حدود 35.6 بهبود روی آنتروپی تصاویر اصلی ارائه می­کند. با این حال این روش خالی از اشکال نیست و می­توان اهداف زیر را پژوهش های آینده دنبال کرد:

* در این مقاله فرض کردیم که جفت تصاویر استریو Rectify شده هستند. در حقیقت فرض کردیم که دوربین های استریو به اندازه­ی کافی به هم نزدیک و سوژه­ی تصویر برداری به اندازه­ی کافی دور است، فرضی که در واقعیت به ندرت رخ می­دهد. بنابراین می­توان ابتدا و پیش از انجام تطابق استریو بلوکی با استخراج ویژگی های یکسان از تصاویر، آن ها را Rectify کرد.
* اندازه­ی بلوک های استفاده شده مقداری ثابت دارد در حالی که منطقی تر به نظر می­رسد که برای نقاط پرفرکانس تصویر از بلوک های کوچکتری استفاده شود تا به خطای کمتری برسیم. این کار را می­توان با تعریف کردن یک آستانه روی مقدار SAD بهترین تطبیق ممکن برای یک بلوک تعریف کرد ولی ساختار ماتریس Disparity باید تغییر یابد چرا که دیگر اندازه­ی همه­ی بلوک ها برابر نیست.
* زمان اجرای الگوریتم AdaGAP بسیار بالاست که این امر آن را برای مصارف Real-time و آنلاین بلااستفاده می­کند.
* با گسترش پنجره­ی تطبیق در راستای طول تصویر، می­توان از این روش برای تصاویری که از یک سوژه و در زمان ها و مکان های کاملاً متفاوت، با دوربین های متفاوت گرفته شده اند استفاده کرد. این کار می­تواند خصوصاً زمانی که لازم باشد تصاویر گرفته شده بر روی یک سرویس ابری ذخیره شوند، موثر واقع شود.

سپاسگزاري

از دکتر نادر کریمی بابت زحمات و راهنمایی­های دلسوزانه و بی دریغشان بی نهایت سپاسگزارم.

مراجع

Avramović, Aleksej, and Branimir Reljin. “Gradient Edge Detection Predictor for Image Lossless Compression.” *Proceedings ELMAR-2010*, 2010, pp. 131–34.

Emlek, Alper, and Murat Peker. “Refinement of Matching Costs for Stereo Disparities Using Recurrent Neural Networks.” *EURASIP Journal on Image and Video Processing*, vol. 2021, no. 1, Apr. 2021, p. 11. *BioMed Central*, <https://doi.org/10.1186/s13640-021-00551-9>.

Poggi, Matteo, et al. “On the Confidence of Stereo Matching in a Deep-Learning Era: A Quantitative Evaluation.” *ArXiv:2101.00431 [Cs]*, Mar. 2021. *arXiv.org*, <http://arxiv.org/abs/2101.00431>.

Stankiewicz, Olgierd, et al. “Chapter 1 - Multiview Video: Acquisition, Processing, Compression, and Virtual View Rendering.” *Academic Press Library in Signal Processing, Volume 6*, edited by Rama Chellappa and Sergios Theodoridis, Academic Press, 2018, pp. 3–74. *ScienceDirect*, <https://doi.org/10.1016/B978-0-12-811889-4.00001-4>.

“Sum of Absolute Differences.” *Wikipedia*, 6 May 2021. *Wikipedia*, <https://en.wikipedia.org/w/index.php?title=Sum_of_absolute_differences&oldid=1021802130>.