پردازش تصویر در دهه آینده
از عمق ریاضیات تا مهندس حرفه‌ای

کتاب ـ جزوه درس کارشناسی ارشد و آزمایشگاه پژوهش‌محور

نهم فصل

کلاسیک بازشناسی و تطبیق ویژگی،
راستی‌آزمایی ،ORB و SIFT تا مقیاس فضای و گوشه قالب، تطبیق از
آموختنی تطبیق به گذار و HOG–SVM بصری، کلمات مقاوم،

آن از که می‌شود متصل کلاس یا شیء نقشه، حافظه، به هنگامی تصویر
فصل این کنیم. استخراج تصمیم قابل و تطبیق قابل تکرار، قابل نمایش
می‌آورد. گرد مشترک لوله خط یک در را الگوریتم و آمار هندسه،

نسخه تفصیلی درس ـ تابستان ۱۴۰۵ / ۲۰۲۶

قابل کامپایل با TeX Live + TeXstudio + XeLaTeX

یادداشت روش‌شناختی فصل نهم

در فصل‌های پیشین آموختیم تصویر چگونه تشکیل، نویزی، بازسازی و قطعه‌بندی می‌شود. اکنون پرسش عوض می‌شود: کدام بخش تصویر ارزش به‌یادسپاری دارد، چگونه همان ساختار را زیر تغییر نور، چرخش، مقیاس و دیدگاه دوباره پیدا کنیم، و چگونه از مجموعه این تطابق‌ها برای بازشناسی شیء یا صحنه استفاده کنیم؟ پاسخ تاریخی بینایی ماشین، خط لوله‌ای چندمرحله‌ای است:

بازشناسی ⟶ راستی‌آزمایی هندسی ⟶ تطبیق ⟶ توصیف ⟶ آشکارسازی.

هر مرحله فرضی را به مرحله بعد تحویل می‌دهد. اگر آشکارساز تکرارپذیر نباشد، بهترین توصیفگر نیز شکست می‌خورد؛ اگر توصیفگر متمایز نباشد، نزدیک‌ترین همسایه پر از تطابق کاذب است؛ اگر هندسه آزموده نشود، شباهت بافتی به‌اشتباه به وجود یک شیء تعبیر می‌شود؛ و اگر تجمیع ویژگی‌ها هندسه و فراوانی را نابود کند، بازشناسی ضعیف خواهد شد.

هدف فصل

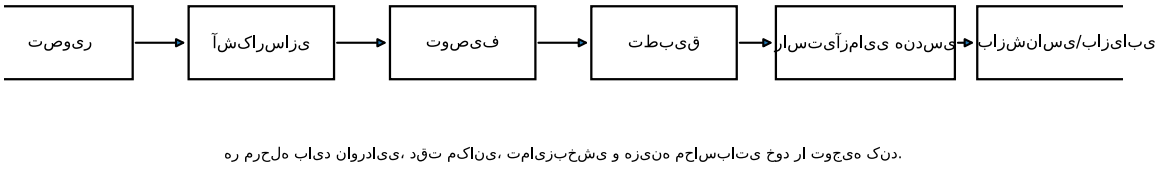
هدف فصل حفظ چهار کمیت در تمام بحث‌هاست:

۱. **هم‌وردایی و ناوردایی:** ویژگی زیر تبدیل تصویر چگونه جابه‌جا یا ثابت می‌ماند؟

۲. **تکرارپذیری و دقت مکانی:** آیا همان ساختار دوباره یافت می‌شود و مکان آن چقدر خطا دارد؟

۳. **تمایزبخشی و احتمال ابهام:** یک توصیفگر چند وصله نامرتبط را مشابه می‌بیند؟

۴. **هزینه و مقیاس‌پذیری:** استخراج، جست‌وجو و راستی‌آزمایی برای هزار تصویر یا یک میلیارد ویژگی چقدر هزینه دارد؟



هر مرحله باید نوآوری، دقت مکانی، تمایزبخشی و هزینه محاسباتی خود را توجیه کند.

شکل ۱: خط لوله کلاسیک ویژگی تا بازشناسی. روش‌های جدید بخشی از این بلوک‌ها را می‌آموزند یا ادغام می‌کنند، اما پرسش‌های بنیادی آن‌ها پابرجاست.

نقشه فشرده فصل

محور	پرسش و خروجی آموزشی
مدل تبدیل	روشنایی، نویز، چرخش، مقیاس، آفین و پرسپکتیو چگونه بر وصله اثر می‌گذارند؟
تطبیق قالب	SSD، SAD، NCC ZNCC و correlation phase چه فرض‌هایی دارند و چگونه سریع می‌شوند؟
نقطه کلیدی	Moravec، Harris، Shi–Tomasi FAST و AGAST چه ساختاری را آشکار می‌کنند؟
فضای مقیاس	Gaussian، scale-space، LoG، DoG Hessian و affine adaptation چگونه ناوردایی می‌سازند؟
توصیفگر	SIFT، RootSIFT، HOG context shape و توصیفگرهای دودویی چه اطلاعاتی را نگه می‌دارند؟
تطبیق	فاصله اقلیدسی، کسینوسی، ماهالانوبیس و همینگ؛ ratio test، cross-check و assignment چگونه انتخاب می‌شوند؟
راستی‌آزمایی	RANSAC، PROSAC LO-RANSAC و MAGSAC++ چگونه مدل هندسی را از برون‌ریزها جدا می‌کنند؟
بازشناسی مستقیم	generalized template، Hough، chamfer، PCA/LDA، kNN، SVM و boosting چه زمانی مناسب‌اند؟
واژگان بصری	BoVW، spatial TF–IDF، VLAD pyramid و Fisher Vector چگونه ویژگی‌های محلی را به بردار تصویر تبدیل می‌کنند؟
ارزیابی	matching repeatability، pose score، AUC، mAP confusion و آزمون تاب‌آوری چگونه گزارش می‌شوند؟
مرز دانش	SuperPoint، LightGlue، LoFTR، RoMa، OmniGlue RDD و تطبیق گرهای ۲۰۲۶ چه چیزی را تغییر داده‌اند؟
مهندسی	چگونه خط لوله reproducible، سریع، بدون leakage و دارای failure taxonomy ساخته شود؟

فهرست مطالب

یادداشت روش‌شناختی فصل نهم آ

نقشه فشرده فصل پ

۹ ویژگی، تطبیق و بازشناسی کلاسیک ۱

۱.۹ ویژگی چیست؟ از مقدار پیکسل تا ساختار قابل تطبیق ۱

۱.۱.۹ گروه تبدیل و هزینه ناوردایی ۲

۲.۱.۹ معیارهای کیفیت نقطه و ناحیه ۲

۲.۹ تطبیق قالب: ساده، بنیادی و هنوز کاربردی ۳

۱.۲.۹ چرا همبستگی نرمال‌شده؟ ۳

۲.۲.۹ محاسبه سریع با انتگرال و FFT ۴

۳.۹ آشکارسازی گوشه: از انرژی تغییر وصله تا مقادیر ویژه ۵

۱.۳.۹ تفسیر طیفی ۵

۲.۳.۹ FAST و AGAST: تصمیم روی دایره ۶

۴.۹ فضای مقیاس: چرا یک نقطه باید اندازه داشته باشد؟ ۷

۱.۴.۹ نرمال‌سازی مشتق ۷

۲.۴.۹ DoG به‌عنوان تقریب LoG ۸

۳.۴.۹ مکان‌یابی زیرپیکسلی و حذف پاسخ لبه ۸

۴.۴.۹ Hessian و detector blob ۹

۵.۴.۹ هم‌وردایی آفین و نواحی پایدار ۹

۵.۹ تخصیص جهت و نرمال‌سازی ناحیه ۱۰

۶.۹ توصیفگرهای محلی: چه چیزی از وصله باقی بماند؟ ۱۰

۱.۶.۹ وصله خام، whitening و NCC ۱۱

۲.۶.۹ histogram SIFT: جهت با pooling مکانی ۱۱

۳.۶.۹ RootSIFT و هندسه Hellinger ۱۲

۴.۶.۹ HOG: توصیف dense برای بازشناسی ۱۳
۵.۶.۹ Context Shape و descriptor شکل ۱۳
۶.۶.۹ توصیفگرهای دودویی ۱۳
۷.۹ تطبیق توصیفگر: از فاصله تا correspondence ۱۵
۱.۷.۹ انتخاب معیار فاصله ۱۵
۲.۷.۹ آزمون نسبت Lowe ۱۵
۳.۷.۹ ،neighbor nearest mutual cross-check و uniqueness ۱۶
۴.۷.۹ جست‌وجوی دقیق و تقریبی ۱۶
۸.۹ راستی‌آزمایی هندسی: شباهت توصیفگر کافی نیست ۱۸
۱.۸.۹ RANSAC و احتمال موفقیت ۱۸
۲.۸.۹ threshold و مدل نویز ۱۹
۳.۸.۹ پیشرفت‌های RANSAC ۲۰
۴.۸.۹ تباهی و انتخاب مدل ۲۰
۹.۹ از match به بازشناسی: سه خانواده کلاسیک ۲۰
۱.۹.۹ تطبیق مستقیم قالب، chamfer و Hough generalized ۲۱
۲.۹.۹ ویژگی‌های کلی شکل و ممان ۲۱
۳.۹.۹ PCA و Eigenfaces ۲۱
۴.۹.۹ LDA و Fisherfaces ۲۲
۱۰.۹ طبقه‌بندهای کلاسیک روی ویژگی ۲۲
۱.۱۰.۹ kNN و هندسه فضای ویژگی ۲۳
۲.۱۰.۹ مدل مولد گاوسی: LDA و QDA به‌عنوان classfiier ۲۳
۳.۱۰.۹ ماشین بردار پشتیبان ۲۳
۴.۱۰.۹ چندکلاسه و calibration ۲۴
۵.۱۰.۹ AdaBoost و cascade ۲۴
۱۱.۹ بازنمایی کلمات بصری: از ویژگی محلی به بردار تصویر ۲۵
۱.۱۱.۹ ساخت واژگان ۲۵
۲.۱۱.۹ ،soft hard و assignment multiple ۲۶
۳.۱۱.۹ TF–IDF و فهرست معکوس ۲۶
۴.۱۱.۹ burstiness و words stop ۲۷
۵.۱۱.۹ Matching Pyramid Spatial ۲۷
۶.۱۱.۹ VLAD: تجمیع residual ۲۸

۷.۱۱.۹ Vector: Fisher گرادیان log-likelihood ۲۹
۱۲.۹ بازشناسی کلاسیک شیء و چهره: خط لوله کامل ۲۹
۱.۱۲.۹ SVM + HOG برای آشکارسازی شیء ۲۹
۲.۱۲.۹ Model Part Deformable ۳۰
۳.۱۲.۹ Fisherfaces Eigenfaces، و زیرفضاهای چهره ۳۰
۴.۱۲.۹ Histogram Pattern Binary Local ۳۱
۱۳.۹ یادگیری متریک و بازشناسی مبتنی بر فاصله ۳۱
۱۴.۹ تطبیق چندتصویری، مسیر و مکان ۳۱
۱.۱۴.۹ retrieval سپس localization ۳۲
۱۵.۹ روش‌های آموختنی: ادامه یا جایگزین خط لوله کلاسیک؟ ۳۲
۱.۱۵.۹ SuperPoint: آشکارسازی و توصیف sefl-supervised ۳۲
۲.۱۵.۹ SuperGlue و LightGlue ۳۲
۳.۱۵.۹ LoFTR و matching detector-free ۳۳
۴.۱۵.۹ matching Dense و ویژگی‌های بنیادی ۳۳
۱۶.۹ ارزیابی آشکارساز و تطبیق گر ۳۴
۱.۱۶.۹ repeatability ۳۴
۲.۱۶.۹ matching، recall precision و score ۳۴
۳.۱۶.۹ خطای هندسی و AUC pose ۳۵
۴.۱۶.۹ ارزیابی بازشناسی ۳۵
۵.۱۶.۹ پروتکل robustness ۳۵
۱۷.۹ آزمایشگاه کدنویسی فصل ۳۶
۱.۱۷.۹ Harris از صفر با قرارداد نوع داده ۳۶
۲.۱۷.۹ RANSAC خط و تعمیم به homography ۳۷
۳.۱۷.۹ SVM + BoVW در pipeline امن ۳۸
۱۸.۹ طراحی آزمایش‌های عمیق و بازتولیدپذیر ۳۹
۱.۱۸.۹ آزمایش ۱: ناوردایی کنترل‌شده ۴۰
۲.۱۸.۹ آزمایش ۲: threshold و trade-off ۴۰
۳.۱۸.۹ آزمایش ۳: budget descriptor ۴۰
۴.۱۸.۹ آزمایش ۴: BoVW، VLAD و FV ۴۰
۵.۱۸.۹ آزمایش ۵: کلاسیک در برابر آموختنی ۴۰
۶.۱۸.۹ آزمایش ۶: shfit domain ۴۰
۱۹.۹ خطاهای رایج در مقاله و پروژه ۴۱

۲۰.۹ پرسش‌های تصمیم مهندسی ۴۱
۲۱.۹ تمرین‌های نظری و اثباتی ۴۲
۲۲.۹ پروژه‌های پایان فصل ۴۳
۱.۲۲.۹ پروژه A: آزمایشگاه ویژگی کلاسیک ۴۳
۲.۲۲.۹ پروژه B: موتور تطبیق و panorama ۴۳
۳.۲۲.۹ پروژه C: بازشناسی کلاسیک کامل ۴۳
۴.۲۲.۹ پروژه D: بازیابی تصویر مقیاس‌پذیر ۴۳
۵.۲۲.۹ پروژه E: کلاسیک در برابر LightGlue/LoFTR ۴۳
۶.۲۲.۹ پروژه F: open-set چهره یا شیء ۴۳
۲۳.۹ پرامپت پیشنهادی برای Codex ۴۳
۲۴.۹ تقسیم پیشنهادی ویدئوهای ۱۵ دقیقه‌ای ۴۵
۲۵.۹ جمع‌بندی و پل به فصل دهم ۴۷

منابع منتخب فصل **۴۹**

راهنمای کامپایل، فونت و اجرای آزمایش‌ها **۵۳**

فصل ۹

ویژگی، تطبیق و بازشناسی کلاسیک

اهداف یادگیری

در پایان فصل، دانشجو باید بتواند:

- پاسخ آشکارساز Harris، LoG/DoG و Hessian را از مدل تغییر وصله استخراج کند؛
- مفهوم هم‌وردایی آشکارساز و ناوردایی توصیفگر را با زبان گروه تبدیل توضیح دهد؛
- SIFT و HOG را نه به‌صورت نام، بلکه به‌صورت histogram جهت‌دار، نرمال‌سازی و pooling پیاده‌سازی کند؛
- برای توصیفگرهای حقیقی و دودویی معیار فاصله و جست‌وجوی مناسب انتخاب کند؛
- تعداد iteration لازم RANSAC را محاسبه و degeneracy و threshold را تحلیل کند؛
- spatial BoVW، VLAD pyramid و Vector Fisher را از های descriptor محلی بسازد؛
- HOG+SVM، Eigenfaces/Fisherfaces و cascade AdaBoost را آموزش و ارزیابی کند؛
- روش کلاسیک را با SuperPoint/LightGlue/LoFTR در پروتکل هندسی عادلانه مقایسه کند؛
- بین دقت تطبیق، پوشش، سرعت، حافظه، قابلیت توضیح و robustness تصمیم مهندسی بگیرد.

۱۰۹ ویژگی چیست؟ از مقدار پیکسل تا ساختار قابل تطبیق

فرض کنید تصویر مشاهده‌شده از یک صحنه مرجع با تبدیل هندسی g، تبدیل فوتومتریک ϕ و نویز n ساخته شود:

$$I_2(\boldsymbol{u}) = \phi\big(I_1(g^{-1}\boldsymbol{u})\big) + n(\boldsymbol{u}). \tag{۱۰۹}$$

هدف ویژگی، ساخت نگاشتی Φ است که اطلاعات مورد نیاز وظیفه را حفظ و تغییرات مزاحم را حذف کند. دو مفهوم باید جدا شوند.

تعریف ۱.۹ (هم‌وردایی آشکارساز). اگر آشکارساز D مجموعه نواحی یا نقاط را برگرداند، نسبت به تبدیل g هم‌ورداست هرگاه

$$D(I \circ g^{-1}) = gD(I).$$

یعنی مکان ویژگی همراه تصویر تبدیل می‌شود، نه اینکه ثابت بماند.

تعریف ۲.۹ (ناوردایی توصیفگر). توصیفگر Ψ نسبت به g ناورداست اگر پس از نرمال‌سازی ناحیه

$$\Psi(I, \boldsymbol{p}) \approx \Psi(I \circ g^{-1}, g\boldsymbol{p}).$$

برابری دقیق معمولاً ناممکن و حتی نامطلوب است؛ تقریب باید تغییرات مجاز را تحمل و تفاوت ساختارهای واقعاً متفاوت را حفظ کند.

اصل راهنما

آشکارساز باید **کجا** را پایدار پیدا کند؛ توصیفگر باید بگوید **چه چیزی** در آنجاست؛ matcher باید شباهت را به فرضیه correspondence تبدیل کند؛ و هندسه باید بررسی کند آیا مجموعه فرضیه‌ها می‌تواند از یک دوربین یا تبدیل مشترک آمده باشد.

۱.۱.۹ گروه تبدیل و هزینه ناوردایی

تبدیل‌های رایج به‌صورت زیر مرتب می‌شوند:

$$\text{انتقال:} \quad \boldsymbol{u}' = \boldsymbol{u} + \boldsymbol{t}, \tag{۲.۹}$$

$$\text{تشابه:} \quad \boldsymbol{u}' = s\boldsymbol{R}\boldsymbol{u} + \boldsymbol{t}, \tag{۳.۹}$$

$$\text{آفین:} \quad \boldsymbol{u}' = \boldsymbol{A}\boldsymbol{u} + \boldsymbol{t}, \tag{۴.۹}$$

$$\text{پروژکتیو:} \quad \widetilde{\boldsymbol{u}}' \sim \boldsymbol{H}\widetilde{\boldsymbol{u}}. \tag{۵.۹}$$

هر ناوردایی هزینه دارد. توصیفگری که نسبت به همه چرخش‌ها و بازتاب‌ها ناوردا باشد ممکن است حروف «ب» و «پ» یا دو قطعه صنعتی جهت‌دار را بیش از حد مشابه کند. بنابراین ناوردایی باید با کاربرد تعریف شود، نه با شعار «هرچه بیشتر بهتر».

۲.۱.۹ معیارهای کیفیت نقطه و ناحیه

برای یک آشکارساز، چهار معیار اصلی‌اند:

- **repeatability:** سهم ساختارهای قابل مشاهده در دو تصویر که هر دو بار آشکار شده‌اند؛
- **localization error:** فاصله نقطه یا اختلاف هم‌پوشانی ناحیه پس از انتقال با هندسه حقیقی؛

- **coverage:** آیا نقاط فقط روی یک بافت پرتکرار جمع شده‌اند یا کل تصویر را می‌پوشانند؟
- **stability:** حساسیت مکان، مقیاس و جهت به نویز و تغییر نمونه‌برداری.

برای نواحی بیضوی A و B، خطای هم‌پوشانی متداول از نسبت اشتراک به اجتماع ساخته می‌شود:

$$\epsilon_{\text{overlap}} = ۱ - \frac{|A \cap B|}{|A \cup B|}. \tag{۶.۹}$$

این معیار وابسته به آستانه هم‌پوشانی است و باید همراه تعداد نواحی و محدوده دید مشترک گزارش شود [۴، ۵].

خطای رایج

تعداد زیاد keypoint نشانه کیفیت نیست. یک آشکارساز می‌تواند هزار نقطه روی بافت تکراری تولید کند و برای تخمین وضعیت دوربین از آشکارسازی با دویست نقطه متنوع بدتر باشد. همیشه repeatability، توزیع مکانی، تعداد inlier هندسی و هزینه را همزمان گزارش کنید.

۲.۹ تطبیق قالب: ساده، بنیادی و هنوز کاربردی

پیش از نقاط کلیدی، مسئله کلاسیک یافتن یک قالب T در تصویر I است. برای جابه‌جایی $\boldsymbol{d}$، مجموع مربعات اختلاف

$$E_{\text{SSD}}(\boldsymbol{d}) = \sum_{\boldsymbol{x} \in \Omega} [I(\boldsymbol{x} + \boldsymbol{d}) - T(\boldsymbol{x})]^۲ \tag{۷.۹}$$

است. اگر نویز جمعی گاوسی مستقل باشد، کمینه‌کردن SSD همان بیشینه‌کردن درست‌نمایی است. برای نویز لاپلاس، SAD طبیعی‌تر است:

$$E_{\text{SAD}}(\boldsymbol{d}) = \sum_{\boldsymbol{x} \in \Omega} |I(\boldsymbol{x} + \boldsymbol{d}) - T(\boldsymbol{x})| \,. \tag{۸.۹}$$

۱.۲.۹ چرا همبستگی نرمال‌شده؟

اگر $I' = aI + b$ باشد، SSD به روشنایی و contrast حساس است. ضریب همبستگی صفرمیانگین نرمال‌شده

$$\text{ZNCC}(\boldsymbol{d}) = \frac{\sum_\Omega (T - \bar{T})(I_{\boldsymbol{d}} - \bar{I}_{\boldsymbol{d}})}{\sqrt{\sum_\Omega (T - \bar{T})^۲} \sqrt{\sum_\Omega (I_{\boldsymbol{d}} - \bar{I}_{\boldsymbol{d}})^۲} + \varepsilon} \tag{۹.۹}$$

نسبت به تبدیل مثبت آفین شدت ناورداست. این نتیجه از حذف میانگین و تقسیم بر انحراف معیار می‌آید. با این حال، ZNCC در وصله‌های تقریباً ثابت بدشرط است و در occlusion شدید همچنان شکست می‌خورد.

۲.۲.۹ محاسبه سریع با انتگرال و FFT

بسط SSD می‌دهد:

$$E_{\mathrm{SSD}}(\boldsymbol{d}) = \sum I_{\boldsymbol{d}}^{۲} + \sum T^{۲} - ۲ \sum I_{\boldsymbol{d}} T. \tag{۱۰.۹}$$

دو جمله نخست با تصویر انتگرالی و جمله همبستگی با convolution یا FFT محاسبه می‌شود. برای انتقال خالص، قضیه شیفت فوریه پایه correlation phase است:

$$I_{۲}(\boldsymbol{x}) = I_{۱}(\boldsymbol{x} - \boldsymbol{d}) \quad \Longrightarrow \quad \widehat{I}_{۲}(\boldsymbol{\omega}) = e^{-j\boldsymbol{\omega}^{\mathsf{T}}\boldsymbol{d}} \widehat{I}_{۱}(\boldsymbol{\omega}). \tag{۱۱.۹}$$

با نرمال‌سازی cross-power spectrum، تبدیل معکوس قله‌ای در $\boldsymbol{d}$ تولید می‌کند:

$$R(\boldsymbol{\omega}) = \frac{\widehat{I}_{۱}(\boldsymbol{\omega}) \widehat{I}_{۲}^{*}(\boldsymbol{\omega})}{\left| \widehat{I}_{۱}(\boldsymbol{\omega}) \widehat{I}_{۲}^{*}(\boldsymbol{\omega}) \right| + \varepsilon}. \tag{۱۲.۹}$$

کاربرد مهندسی: ثبت تصویر و کنترل کیفیت

Template matching برای قطعات با وضعیت تقریباً ثابت، fiducialها، کنترل چاپ، ردیابی کوتاه‌مدت و ثبت translation-only هنوز گزینه‌ای شفاف و سریع است. شکست اصلی آن تغییر مقیاس، دوران، تغییر شکل، پس‌زمینه پرتکرار و تغییر نور موضعی است.

کارگاه کد و آزمایش: پیاده‌سازی ZNCC

```python
import numpy as np

def zncc(template: np.ndarray, patch: np.ndarray, eps: float = 1e-8) -> float:
    """Zero-mean normalized cross correlation for equal-size arrays."""
    if template.shape != patch.shape:
        raise ValueError("template and patch must have the same shape")
    t = template.astype(np.float64) - np.mean(template)
    p = patch.astype(np.float64) - np.mean(patch)
    denom = np.linalg.norm(t) * np.linalg.norm(p)
    if denom < eps:
        return float("nan")  # constant patches are not informative
    return float(np.sum(t * p) / denom)
```

در آزمایش، روشنایی $I' = aI + b$، نویز، occlusion blur، و دوران را جداگانه تغییر دهید. نمودار score باید نشان دهد کدام تغییر در مدل ناوردایی ZNCC است و کدام نیست.

۳.۹ آشکارسازی گوشه: از انرژی تغییر وصله تا مقادیر ویژه

برای وصله w و انتقال کوچک $\boldsymbol{\delta} = (u, v)^\top$، تغییر شدت را تعریف می‌کنیم:

$$E(u, v) = \sum_{\boldsymbol{x}} w(\boldsymbol{x}) \left[I(\boldsymbol{x} + \boldsymbol{\delta}) - I(\boldsymbol{x}) \right]^2 . \tag{۱۳.۹}$$

با تقریب تیلور مرتبه اول

$$I(\boldsymbol{x} + \boldsymbol{\delta}) - I(\boldsymbol{x}) \approx I_x u + I_y v$$

و بنابراین

$$E(u, v) \approx \begin{bmatrix} u & v \end{bmatrix} \underbrace{\sum_{\boldsymbol{x}} w(\boldsymbol{x}) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}}_{\boldsymbol{M}} \begin{bmatrix} u \\ v \end{bmatrix} . \tag{۱۴.۹}$$

ماتریس $\boldsymbol{M}$ همان تانسور ساختار فصل هفتم است، اما اینجا نقش آن تصمیم درباره نقطه قابل تطبیق است.

۱.۳.۹ تفسیر طیفی

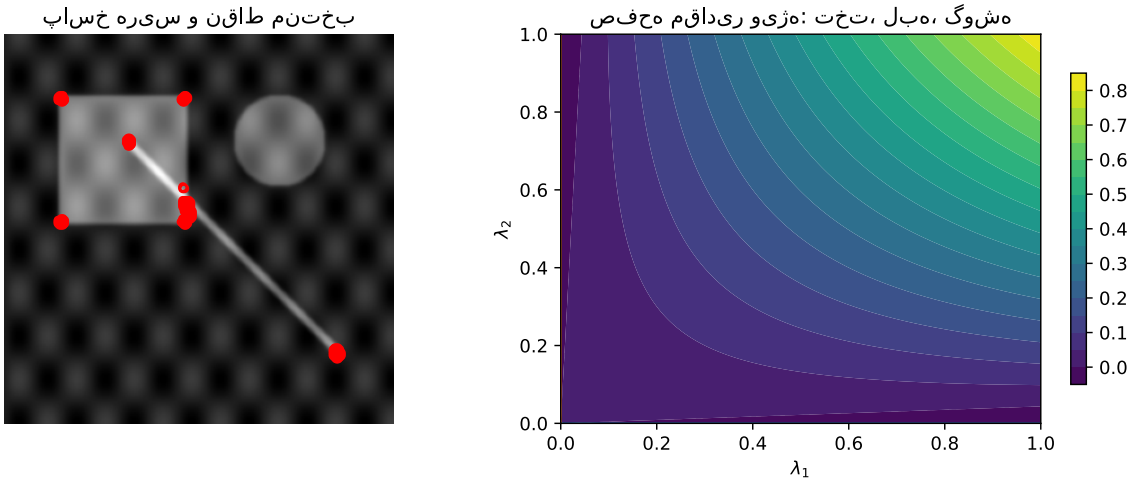
اگر مقادیر ویژه λ_1, λ_2 باشند:

- هر دو کوچک: وصله تخت؛
- یکی بزرگ و دیگری کوچک: لبه؛ تغییر فقط عمود بر لبه قابل تشخیص است؛
- هر دو بزرگ: گوشه یا بافت دوبعدی؛ انتقال در هر جهت انرژی را تغییر می‌دهد.

Harris از محاسبه صریح eigenvalue اجتناب می‌کند:

$$R_H = \det(\boldsymbol{M}) - k \operatorname{tr}(\boldsymbol{M})^2 = \lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2. \tag{۱۵.۹}$$

Shi–Tomasi معیار $\min(\lambda_1, \lambda_2)$ را پیشنهاد می‌کند و مستقیماً ضعیف‌ترین جهت را می‌سنجد [۱، ۲].



شکل ۱۰۹: چپ: نقاط با پاسخ Harris زیاد. راست: پاسخ در صفحه مقادیر ویژه؛ گوشه هنگامی پدید می‌آید که هر دو eigenvalue بزرگ باشند.

الگوریتم ۱۰۹ ـ Harris با suppression non-maximum

۱. تصویر را در مقیاس مشتق σ_d هموار و I_x, I_y را محاسبه کنید.

۲. I_x^2, I_y^2, I_xI_y را در مقیاس integration σ_i هموار کنید.

۳. برای هر پیکسل $R_H = \det M - k(\operatorname{tr} M)^2$ را بسازید.

۴. نقاط با $R_H > \tau$ را نگه دارید.

۵. در همسایگی شعاع r فقط بیشینه محلی را حفظ کنید.

۶. برای دقت زیرپیکسلی، یک تابع درجه دوم روی پاسخ اطراف قله برازش کنید.

پیچیدگی با فیلتر separable و NMS محلی $O(N)$ است. آستانه مطلق بین تصاویر قابل انتقال نیست؛ quantile یا budget ثابت نقاط برای آزمایش منصفانه مناسب‌تر است.

۲.۳.۹ FAST و AGAST: تصمیم روی دایره

FAST به‌جای تانسور، شانزده پیکسل روی دایره Bresenham شعاع ۳ را آزمون می‌کند. نقطه گوشه است اگر حداقل n پیکسل متوالی از مرکز به اندازه t روشن‌تر یا تاریک‌تر باشند. تصمیم درختی ترتیب آزمون‌ها را برای رد سریع نامزدها می‌آموزد. FAST بسیار سریع است، اما مقیاس و جهت را ذاتاً فراهم نمی‌کند. ORB با هرم، orientation و NMS آن را به آشکارساز کامل‌تری تبدیل می‌کند [۳، ۱۲]. AGAST درخت تصمیم را برای معماری‌ها و الگوهای مختلف تعمیم می‌دهد.

خطای رایج

آستانه FAST یا Harris را روی تصویر ۸uint و float با یک عدد ثابت استفاده نکنید. مقیاس gamma intensity، و preprocessing مستقیماً پاسخ را عوض می‌کند. قرارداد داده باید در API آشکارساز صریح باشد.

۴.۹ فضای مقیاس: چرا یک نقطه باید اندازه داشته باشد؟

یک گوشه در تصویر نزدیک ممکن است در تصویر دور به لکه‌ای کوچک تبدیل شود. آشکارسازی در یک مقیاس ثابت نمی‌تواند زیر zoom تکرارپذیر باشد. فضای مقیاس خطی با convolution گاوسی تعریف می‌شود:

$$L(\boldsymbol{x},\sigma) = G(\boldsymbol{x};\sigma) * I(\boldsymbol{x}), \qquad G(\boldsymbol{x};\sigma) = \frac{۱}{۲\pi\sigma^۲}\exp\left(-\frac{\|\boldsymbol{x}\|^۲}{۲\sigma^۲}\right). \tag{۱۶.۹}$$

گاوسی تحت اصول خطی‌بودن، ناوردایی انتقال، نیم‌گروه و عدم ایجاد ساختار جدید، هسته طبیعی فضای مقیاس است. خاصیت نیم‌گروه

$$G(\cdot;\sigma_۱) * G(\cdot;\sigma_۲) = G\left(\cdot;\sqrt{\sigma_۱^۲+\sigma_۲^۲}\right) \tag{۱۷.۹}$$

اجازه می‌دهد هاoctave به‌صورت بازگشتی ساخته شوند.

۱.۴.۹ نرمال‌سازی مشتق

مقدار مشتق با بزرگ‌شدن σ کاهش می‌یابد. برای مقایسه میان مقیاس‌ها از مشتق نرمال‌شده

$$\partial_{\xi_i} = \sigma^{\gamma}\partial_{x_i} \tag{۱۸.۹}$$

استفاده می‌شود. برای blob دوبعدی، لاپلاسی نرمال‌شده

$$\mathcal{L}(\boldsymbol{x},\sigma) = \sigma^۲\nabla^۲ L(\boldsymbol{x},\sigma) \tag{۱۹.۹}$$

در مقیاس متناسب با اندازه blob اکسترمم دارد. این اصل «انتخاب مقیاس خودکار» است.

۲.۴.۹ DoG به‌عنوان تقریب LoG

از معادله گرما داریم

$$\frac{\partial G}{\partial \sigma} = \sigma \nabla^{۲} G. \tag{۲۰.۹}$$

پس برای k نزدیک یک:

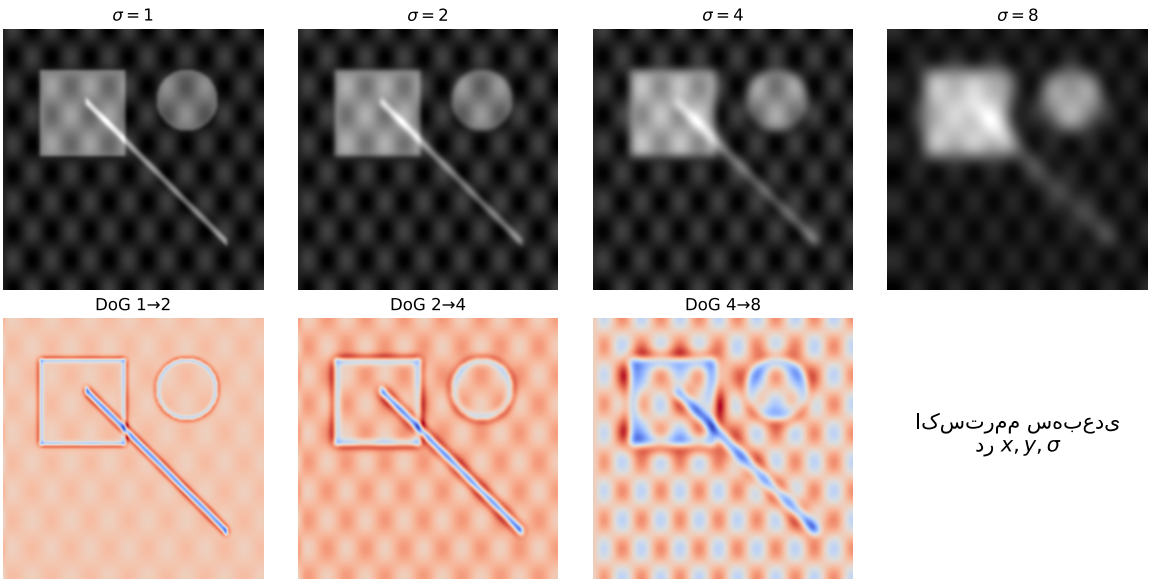
$$G(k\sigma) - G(\sigma) \approx (k-۱)\sigma \frac{\partial G}{\partial \sigma} \tag{۲۱.۹}$$

$$= (k-۱)\sigma^{۲} \nabla^{۲} G. \tag{۲۲.۹}$$

بنابراین Gaussians of Dffierence

$$D(\boldsymbol{x}, \sigma) = L(\boldsymbol{x}, k\sigma) - L(\boldsymbol{x}, \sigma) \tag{۲۳.۹}$$

تقریبی مقیاس‌نرمال از LoG است و با تفریق دو تصویر هموارشده ارزان محاسبه می‌شود. SIFT اکسترمم D را در ۲۶ همسایه فضای x, y, σ می‌یابد [۶].



شکل ۲.۹: ردیف بالا: تصویر در مقیاس‌های افزاینده. ردیف پایین: DoG میان مقیاس‌های مجاور. نقطه مقیاس‌ناوردا باید در فضای سه‌بعدی مکان–مقیاس اکسترمم باشد.

۳.۴.۹ مکان‌یابی زیرپیکسلی و حذف پاسخ لبه

بردار offset اکسترمم با بسط درجه دوم DoG حول نمونه $z = (x, y, \sigma)$ به دست می‌آید:

$$D(z + \Delta z) \approx D(z) + \nabla D^{\mathsf{T}} \Delta z + \frac{۱}{۲} \Delta z^{\mathsf{T}} \boldsymbol{H}_D \Delta z. \tag{۲۴.۹}$$

شرط ایستایی می‌دهد

$$\widehat{\Delta z} = -\boldsymbol{H}_D^{-1} \nabla D. \tag{۲۵.۹}$$

اگر offset بزرگ باشد، نامزد به نمونه مجاور منتقل و دوباره برازش می‌شود. کنتراست پایین با مقدار interpolated حذف می‌شود. برای رد پاسخ‌های کشیده روی لبه، اگر $\alpha \geq \beta$ هایHessian eigenvalue مکانی باشند و $r = \alpha/\beta$، آنگاه

$$\frac{\mathrm{tr}(H)^2}{\det(H)} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(r+1)^2}{r}. \tag{۲۶.۹}$$

نامزدهایی که این نسبت از حد متناظر $r_{\max}$ بزرگ‌تر است رد می‌شوند.

۴.۴.۹ Hessian و detector blob

تصویر Hessian

$$\boldsymbol{H}_L = \begin{bmatrix} L_{xx} & L_{xy} \\ L_{xy} & L_{yy} \end{bmatrix} \tag{۲۷.۹}$$

و پاسخ مقیاس‌نرمال آن

$$R_{\mathrm{Hess}}(\boldsymbol{x}, \sigma) = \sigma^4 \det(\boldsymbol{H}_L) \tag{۲۸.۹}$$

به ساختارهای blob حساس است. SURF پاسخ Hessian را با filter box و تصویر انتگرالی تقریب می‌کند. KAZE و AKAZE به‌جای فضای مقیاس گاوسی خطی، dffiusion غیرخطی را به‌کار می‌گیرند تا مرزها بهتر حفظ شوند؛ AKAZE با Dffiusion Explicit Fast هزینه را کاهش می‌دهد [۹، ۱۰].

۵.۴.۹ هم‌وردایی آفین و نواحی پایدار

تغییر دیدگاه محلی تقریباً آفین است. Harris–Affine و Hessian–Affine شکل بیضی ناحیه را با iteration ماتریس moment second نرمال می‌کنند. MSER مجموعه‌های تراز شدت را دنبال می‌کند و نواحی‌ای را برمی‌گزیند که مساحتشان در بازه threshold کم تغییر کند:

$$q(i) = \frac{|Q_{i+\Delta} \setminus Q_{i-\Delta}|}{|Q_i|}. \tag{۲۹.۹}$$

کمینه‌های $q(i)$ نواحی stable maximally هستند. این روش برای نوشته، نماهای شهری و ساختارهای دارای contrast پایدار مفید است [۸]. ارزیابی جامع نواحی affine-covariant نشان داد که انتخاب detector و descriptor باید جداگانه سنجیده شود [۴، ۵].

الگوریتم ۲.۹ ـ آشکارسازی اکسترمم فضای مقیاس

۱. برای هر octave، $s+3$ تصویر گاوسی با نسبت $k = 2^{1/s}$ بسازید.
۲. تصاویر مجاور را تفریق و $s+2$ لایه DoG تولید کنید.
۳. هر نمونه داخلی را با ۸ همسایه همان مقیاس و ۹+۹ همسایه مقیاس مجاور مقایسه کنید.
۴. نامزد را با معادله (۲۵.۹) زیرپیکسلی کنید.
۵. نامزدهای کنتراست پایین و edge-like را حذف کنید.
۶. مختصات را به دستگاه تصویر اصلی و مقیاس فیزیکی تبدیل کنید.

اگر هاkernel separable باشند، ساخت pyramid نسبت به تعداد پیکسل‌های کل هاoctave خطی است. هزینه اصلی descriptor و matching است، نه صرفاً DoG.

۵.۹ تخصیص جهت و نرمال‌سازی ناحیه

برای ناوردایی دوران، هایgradient اطراف keypoint در مقیاس انتخاب‌شده محاسبه می‌شوند:

$$m(\boldsymbol{x}) = \sqrt{L_x^2 + L_y^2}, \qquad \theta(\boldsymbol{x}) = \operatorname{atan}(L_y, L_x). \tag{۳۰.۹}$$

هر نمونه با وزن گاوسی به histogram جهت رأی می‌دهد. قله اصلی جهت canonical را تعیین می‌کند؛ قله‌های نزدیک به قله اصلی می‌توانند keypoint اضافی بسازند. سپس patch به دستگاه محلی

$$\boldsymbol{x}' = \frac{1}{\sigma}\boldsymbol{R}(-\theta)(\boldsymbol{x} - \boldsymbol{x}_0) \tag{۳۱.۹}$$

منتقل می‌شود. این مرحله آشکارساز هم‌وردا را به region support نرمال تبدیل می‌کند.

خطای رایج

جهت‌گذاری در ساختارهای متقارن ناپایدار است. یک دایره یا بافت isotropic جهت یکتا ندارد. تولید چند orientation پوشش را افزایش می‌دهد، اما تعداد descriptor و تطابق کاذب را نیز زیاد می‌کند. ambiguity باید در طراحی matcher لحاظ شود.

۶.۹ توصیفگرهای محلی: چه چیزی از وصله باقی بماند؟

توصیفگر تابعی از patch نرمال‌شده است:

$$\boldsymbol{d} = \Psi(\mathcal{W}(I; \boldsymbol{p}, \sigma, \theta, \boldsymbol{A})) \in \mathbb{R}^D \quad \text{یا} \quad \{0,1\}^D. \tag{۳۲.۹}$$

سه trade-off بنیادی‌اند: localization در برابر pooling، ناوردایی در برابر تمایزبخشی، و ابعاد در برابر سرعت جست‌وجو.

۱.۶.۹ وصله خام، whitening و NCC

ساده‌ترین descriptor بردار پیکسل‌های وصله است. پس از حذف میانگین و نرمال‌سازی ℓ_2، فاصله اقلیدسی و correlation به هم مرتبط‌اند:

$$\|\hat{p} - \hat{q}\|_2^2 = 2 - 2\hat{p}^\top \hat{q}. \tag{۳۳.۹}$$

whitening با کوواریانس داده

$$z = \Sigma^{-1/2}(p - \mu) \tag{۳۴.۹}$$

جهت‌های پرتکرار و پرهمبستگی را متعادل می‌کند، اما نسبت به misalignment حساس است.

۲.۶.۹ histogram SIFT: جهت با pooling مکانی

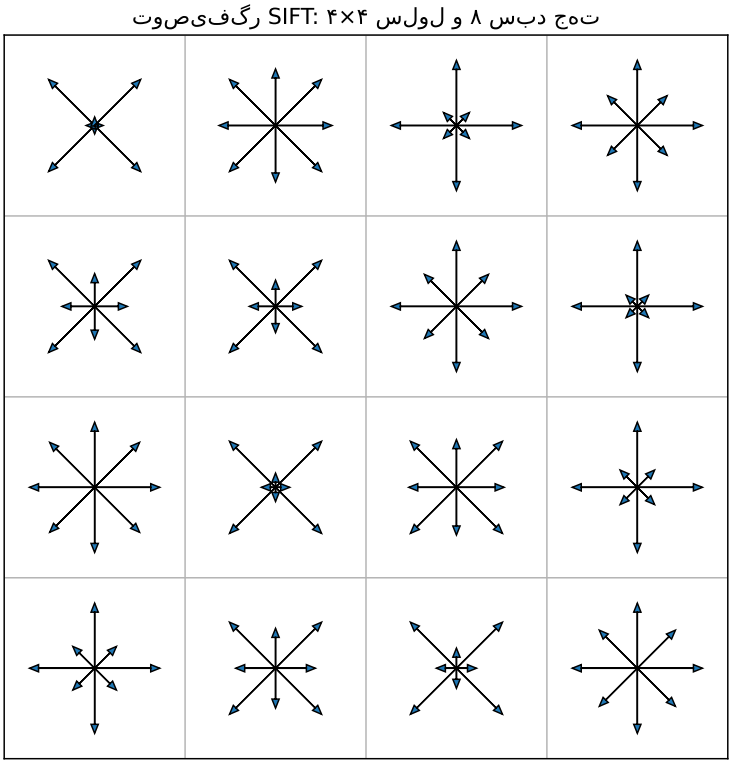
SIFT یک پنجره معمولاً 16×16 را به 4×4 سلول تقسیم و در هر سلول histogram هشت‌جهتی می‌سازد؛ در نتیجه $4 \times 4 \times 8 = 128$ بعد دارد. رأی هر gradient به‌صورت trilinear میان دو سلول مکانی و دو bin جهت پخش می‌شود تا تغییر کوچک مکان یا جهت descriptor را ناگهانی عوض نکند. بردار سپس:

۱. با نرم ℓ_2 نرمال می‌شود؛

۲. مؤلفه‌ها در حدی مانند ۰٫۲ clip می‌شوند؛

۳. دوباره نرمال می‌شود.

clip اثر های gradient بسیار غالب و تغییر illumination موضعی را کاهش می‌دهد.



شکل ۳.۹: نمای آموزشی SIFT descriptor. pooling جهت در سلول‌های مکانی، کمی خطای localization را تحمل می‌کند و در عین حال آرایش محلی را نگه می‌دارد.

مسیر اثبات: چرا histogram نسبت به جابه‌جایی کوچک پایدارتر از پیکسل خام است؟

اگر gradient از مختصات x به $x + \Delta x$ جابه‌جا شود، رأی trilinear به‌طور پیوسته میان سلول‌ها بازتوزیع می‌شود. در patch خام، همان جابه‌جایی می‌تواند مؤلفه‌ای را از یک index به index دیگر منتقل کند و فاصله بزرگی بسازد. pooling یک low-pass در فضای مکان–جهت است؛ پایداری می‌خرد، اما جزئیات فرکانس بالا را حذف می‌کند.

۳.۶.۹ RootSIFT و هندسه Hellinger

اگر histogram غیرمنفی $\boldsymbol{h}$ باشد، RootSIFT ابتدا ℓ_1 نرمال و سپس ریشه مؤلفه‌ای می‌گیرد:

$$\boldsymbol{r}_i = \sqrt{\frac{h_i}{\sum_j h_j + \varepsilon}}. \tag{۳۵.۹}$$

فاصله اقلیدسی میان $\boldsymbol{r}$ها معادل فاصله Hellinger میان هاست histogram. این تغییر ساده اغلب matching و retrieval را بهبود می‌دهد [۱۵].

۴.۶.۹ HOG: توصیف dense برای بازشناسی

HOG همان ایده histogram جهت را به شبکه dense سلول‌ها و normalization block گسترش می‌دهد. برای block بردار $\boldsymbol{v}$، نرمال‌سازی Hys:۲L-

$$\boldsymbol{v} \leftarrow \frac{\boldsymbol{v}}{\sqrt{\|\boldsymbol{v}\|_{۲}^{۲} + \varepsilon^{۲}}}, \quad v_i \leftarrow \min(v_i, c), \quad \boldsymbol{v} \leftarrow \frac{\boldsymbol{v}}{\sqrt{\|\boldsymbol{v}\|_{۲}^{۲} + \varepsilon^{۲}}}. \tag{۳۶.۹}$$

SVM HOG+linear نمونه کلاسیک تبدیل gradient محلی به detector شیء است [۱۶].

۵.۶.۹ Context Shape و descriptor شکل

برای نقطه مرزی p_i، توزیع نسبی سایر نقاط در هایbin log-polar ثبت می‌شود:

$$h_i(k) = \#\{q \neq p_i : q - p_i \in \text{bin}(k)\}. \tag{۳۷.۹}$$

هزینه تطبیق دو histogram با $\chi^{۲}$:

$$C_{ij} = \frac{۱}{۲} \sum_k \frac{(h_i(k) - h_j(k))^{۲}}{h_i(k) + h_j(k) + \varepsilon}. \tag{۳۸.۹}$$

سپس assignment و spline تغییر شکل را برآورد می‌کند. این روش برای contour و بازشناسی شکل کلاسیک مهم است [۱۷].

۶.۶.۹ توصیفگرهای دودویی

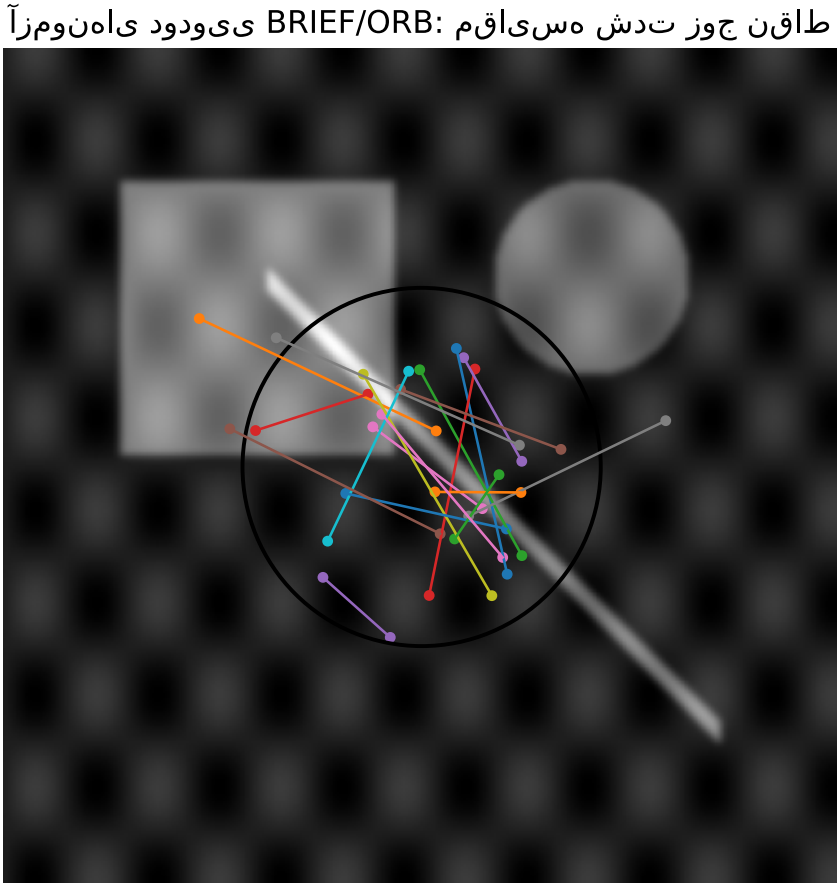
BRIEF مجموعه آزمون‌های شدت است:

$$\tau(\boldsymbol{p}; \boldsymbol{a}_i, \boldsymbol{b}_i) = \begin{cases} ۱, & I(\boldsymbol{p} + \boldsymbol{a}_i) < I(\boldsymbol{p} + \boldsymbol{b}_i), \\ ۰, & \text{otherwise}. \end{cases} \tag{۳۹.۹}$$

رشته D بیتی با XOR و popcount بسیار سریع مقایسه می‌شود:

$$d_{\mathrm{H}}(\boldsymbol{b}, \boldsymbol{c}) = \sum_{i=۱}^{D} b_i \oplus c_i. \tag{۴۰.۹}$$

BRIEF ذاتاً invariant rotation نیست. ORB جهت را با centroid intensity برآورد و الگوی آزمون را می‌چرخاند؛ همچنین زوج آزمون‌های پُرواریانس و کم‌همبستگی را انتخاب می‌کند [۱۱، ۱۲]. BRISK از pattern sampling حلقوی و FREAK از الگوی شبیه شبکیه استفاده می‌کنند. descriptor AKAZE دودویی M-LDB را روی فضای مقیاس غیرخطی می‌سازد.



شکل ۴.۹: نمای شماتیک آزمون‌های زوجی شدت در توصیفگر دودویی. سرعت بالا نتیجه تبدیل descriptor به رشته بیت و فاصله به popcount است.

توصیفگر	نوع	فاصله معمول	نقطه قوت	محدودیت اصلی
SIFT/RootSIFT	حقیقی، ۱۲۸ بعد	ℓ_2/Hellinger	robust و متمایز	هزینه و حافظه بیشتر
HOG	حقیقی، dense	خطی/SVM	شکل و gradient	ناوردایی محدود دیدگاه
BRIEF	دودویی	Hamming	بسیار سریع	بدون جهت/مقیاس ذاتی
ORB	دودویی	Hamming	real-time، آزاد و در دسترس	در تغییر دید شدید ضعیف‌تر از SIFT
BRISK/FREAK	دودویی	Hamming	الگوی نمونه‌برداری کارا	حساس به blur و photometric shfit
AKAZE	دودویی/حقیقی	Hamming/ℓ_2	scale-space لبه‌نگهدار	تنظیمات و هزینه dffiusion

Shape text	Contour/assignment	نیاز به	شکل و formable	de-matching
histogram	χ^2			

۷.۹ تطبیق توصیفگر: از فاصله تا correspondence

برای های descriptor query $\{\boldsymbol{d}_i\}$ و database $\{\boldsymbol{e}_j\}$، تطبیق خام

$$\pi(i) = \arg\min_j d(\boldsymbol{d}_i, \boldsymbol{e}_j) \tag{۴۱.۹}$$

است. اما «نزدیک‌ترین» الزاماً «درست» نیست؛ در هر پایگاه داده یک نزدیک‌ترین همسایه وجود دارد، حتی اگر هیچ correspondence واقعی نباشد.

۱.۷.۹ انتخاب معیار فاصله

$$d_2(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|_2\,, \tag{۴۲.۹}$$

$$d_{\cos}(\boldsymbol{x}, \boldsymbol{y}) = 1 - \frac{\boldsymbol{x}^\top \boldsymbol{y}}{\|\boldsymbol{x}\|\,\|\boldsymbol{y}\|}, \tag{۴۳.۹}$$

$$d_M(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{(\boldsymbol{x} - \boldsymbol{y})^\top \boldsymbol{M} (\boldsymbol{x} - \boldsymbol{y})}, \tag{۴۴.۹}$$

$$d_H(\boldsymbol{b}, \boldsymbol{c}) = \sum_i b_i \oplus c_i. \tag{۴۵.۹}$$

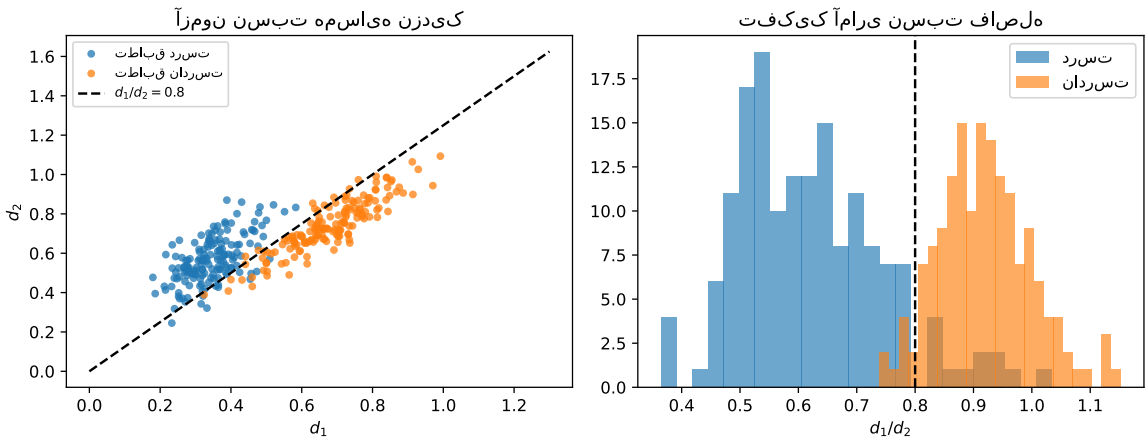
برای های descriptor ℓ_2-نرمال، فاصله اقلیدسی و ranking cosine یکسان می‌دهند. Mahalanobis correlation و اهمیت ابعاد را مدل می‌کند، اما نیازمند برآورد یا یادگیری $\boldsymbol{M} \succeq 0$ است. استفاده Hamming برای float descriptor یا ℓ_2 برای بیت‌ها خطای مفهومی است.

۲.۷.۹ آزمون نسبت Lowe

اگر d_1 و d_2 فاصله نخستین و دومین همسایه باشند، تطابق پذیرفته می‌شود اگر

$$\frac{d_1}{d_2} < \rho. \tag{۴۶.۹}$$

شهود: تطابق درست باید نه‌فقط نزدیک، بلکه نسبت به گزینه جایگزین متمایز باشد. threshold ثابت مثل ۰٫۷ یا ۰٫۸ قانون جهان‌شمول نیست؛ به descriptor، دامنه، تعداد ویژگی و recall مورد نیاز وابسته است. مقاله SIFT این آزمون را برای کاهش clutter background به‌کار برد [۶].



شکل ۵.۹: آزمون نسبت. تطابق درست معمولاً فاصله اول کوچک و فاصله دوم به‌وضوح بزرگ‌تری دارد؛ بافت تکراری دو همسایه تقریباً برابر می‌سازد.

۳.۷.۹ neighbor nearest mutual cross-check، و uniqueness

تطابق mutual اگر

$$\pi_{A\to B}(i) = j, \qquad \pi_{B\to A}(j) = i \tag{۴۷.۹}$$

باشد. این آزمون precision را افزایش و recall را کاهش می‌دهد. در rectfiied stereo می‌توان علاوه بر mutuality، محدودیت disparity و ordering را وارد کرد. در retrieval، یک database descriptor ممکن است به چند query پاسخ دهد؛ assignment یک‌به‌یک را می‌توان با الگوریتم Hungarian روی ماتریس هزینه حل کرد، اما $O(n^3)$ و حساس به points unmatched است. formulation با dummy node یا unbalanced transport optimal انعطاف بیشتری دارد.

۴.۷.۹ جست‌وجوی دقیق و تقریبی

brute-force برای $N_q N_d D$ عملیات ساده و برای مجموعه کوچک قابل اعتماد است. برای مقیاس بزرگ:

- kd-tree در ابعاد پایین مفید، اما در ابعاد زیاد دچار dimensionality of curse می‌شود؛
- kd-trees randomized و k-means hierarchical در FLANN trade-off جست‌وجو/دقت می‌دهند؛
- LSH برای رشته‌های دودویی یا فضاهای metric خاص مناسب است؛
- quantization product بردار را به زیربردار تقسیم و فاصله تقریبی را با table lookup محاسبه می‌کند؛
- file inverted ابتدا cell coarse را انتخاب و سپس PQ را در همان cell اجرا می‌کند.

مستندات OpenCV نیز ۲L را برای SIFT/SURF و Hamming را برای هایdescriptor دودویی تفکیک می‌کند [۴۴].

الگوریتم ۳.۹ ـ تطبیق مقاوم اولیه

۱. نوع descriptor و metric سازگار را تعیین کنید.
۲. برای هر query دو همسایه نزدیک را پیدا کنید.
۳. test ratio با ρ از پیش‌اعلام‌شده اعمال کنید.
۴. در صورت نیاز cross-check mutual را اجرا کنید.
۵. های‌duplicate مکانی یا چند orientation یک keypoint را ادغام کنید.
۶. index، score، مختصات، scale و orientation هر match را نگه دارید؛ فقط تصویر ترسیمی کافی نیست.
۷. تطابق‌ها را به مرحله هندسی تحویل دهید؛ پیش از هندسه نتیجه detection object اعلام نکنید.

کارگاه کد و آزمایش: SIFT/ORB و انتخاب matcher

```python
import cv2 as cv

def detect_and_match(img1, img2, method="sift", ratio=0.75):
    if method == "sift":
        detector = cv.SIFT_create(nfeatures=3000)
        norm = cv.NORM_L2
    elif method == "orb":
        detector = cv.ORB_create(nfeatures=3000, scaleFactor=1.2, nlevels=8)
        norm = cv.NORM_HAMMING
    else:
        raise ValueError("method must be 'sift' or 'orb'")

    k1, d1 = detector.detectAndCompute(img1, None)
    k2, d2 = detector.detectAndCompute(img2, None)
    if d1 is None or d2 is None:
        return k1, k2, []

    matcher = cv.BFMatcher(normType=norm, crossCheck=False)
    pairs = matcher.knnMatch(d1, d2, k=2)
    good = [m for m, n in pairs if m.distance < ratio * n.distance]
    return k1, k2, good
```

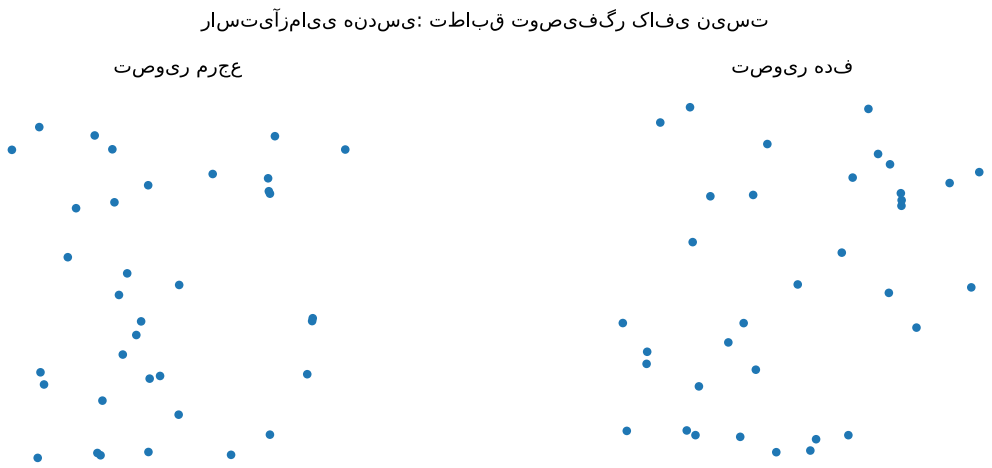
برای گزارش علمی، علاوه بر تعداد good، match تعداد inlier پس از مدل هندسی، reprojection error، زمان detect/describe/match و حافظه descriptor را ذخیره کنید.

۸.۹ راستی‌آزمایی هندسی: شباهت توصیفگر کافی نیست

تطابق‌های اولیه مجموعه‌ای از زوج نقاط $\{(\boldsymbol{x}_i, \boldsymbol{x}'_i)\}$ است. مدل هندسی residual $\boldsymbol{\theta}$

$$r_i(\boldsymbol{\theta}) = d_g(\boldsymbol{x}'_i, f(\boldsymbol{x}_i; \boldsymbol{\theta})) \tag{۴۸.۹}$$

می‌سازد. برای homography از خطای symmetric transfer، برای fundamental matrix از Sampson و برای PnP از reprojection استفاده می‌شود.



شکل ۶.۹: وجود های descriptor match تنها فرضیه می‌سازد. مدل هندسی باید سازگاری جمعی correspondence‌ها را بررسی کند.

۱.۸.۹ RANSAC و احتمال موفقیت

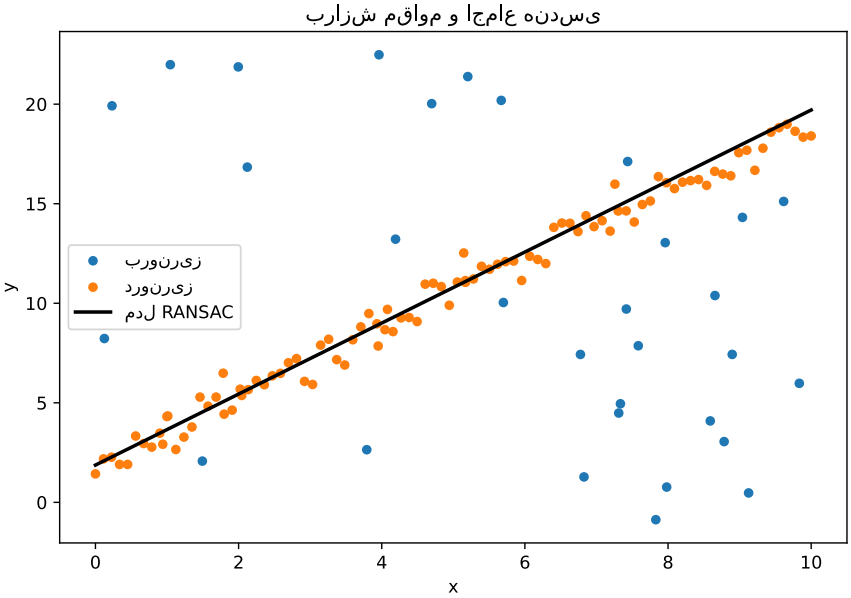
اگر سهم inlier برابر w و اندازه sample minimal برابر s باشد، احتمال اینکه یک sample کاملاً inlier باشد w^s است. پس احتمال شکست پس از N iteration:

$$P_{\text{fail}} = (1 - w^s)^N. \tag{۴۹.۹}$$

برای احتمال موفقیت p:

$$N \geq \frac{\log(1 - p)}{\log(1 - w^s)}. \tag{۵۰.۹}$$

مثلاً با $w = ۰٫۵$، homography با $s = ۴$ و $p = ۰٫۹۹$ حدود ۷۲ iteration می‌خواهد؛ با $w = ۰٫۲$ این عدد به حدود ۲۸۷۶ می‌رسد. بنابراین کیفیت matching اولیه مستقیماً هزینه هندسه را تعیین می‌کند [۱۸].



شکل ۷.۹: RANSAC مدل را از subset حداقلی پیشنهاد و با consensus ارزیابی می‌کند. threshold residual تعریف عملی inlier است.

الگوریتم ۴.۹ ـ RANSAC تطبیقی

۱. $N = N_{\max}$، بهترین score و mask خالی را مقداردهی کنید.

۲. یک sample minimal غیرتباه انتخاب و مدل را برازش کنید.

۳. residual همه داده‌ها را حساب و های inlier $r_i < \tau$ را مشخص کنید.

۴. مدل را با score، robust نه فقط شمارش خام، رتبه‌بندی کنید.

۵. اگر بهتر بود، مدل را روی های inlier پالایش و $\widehat{w}$ را به‌روزرسانی کنید.

۶. N را با معادله (۵.۰.۹) و $\widehat{w}$ کاهش دهید.

۷. پس از توقف، refinement nonlinear روی های inlier اجرا و covariance تقریبی گزارش کنید.

۲.۸.۹ threshold و مدل نویز

آستانه پیکسلی ثابت با تغییر resolution و keypoint uncertainty عادلانه نیست. اگر covariance resid- ual برابر Σ_i باشد، فاصله Mahalanobis

$$r_i^۲ = \boldsymbol{e}_i^\top \Sigma_i^{-۱} \boldsymbol{e}_i \tag{۵.۱.۹}$$

را با quantile توزیع χ^2 مقایسه می‌کنیم. scale، detector localization keypoint و calibra- camera tion باید در uncertainty وارد شوند.

۳.۸.۹ پیشرفت‌های RANSAC

- **PROSAC:** sample را از های‌match با score بهتر آغاز و دامنه را تدریجی گسترش می‌دهد؛
- **LO-RANSAC:** هر مدل خوب را با local optimization روی هاinlier بهبود می‌دهد؛
- **USAC:** sampler، degeneracy verfiication و polishing را در چارچوب واحد ترکیب می‌کند؛
- **MAGSAC++:** به‌جای threshold نویز ثابت، کیفیت و وزن‌ها را با marginalization روی noise scale و IRLS می‌سازد [۲۱].

۴.۸.۹ تباهی و انتخاب مدل

چهار نقطه تقریباً هم‌خط برای homography تباه‌اند؛ نقاط روی صفحه خاص ممکن است fundamental matrix را با homography اشتباه‌پذیر کنند؛ حرکت دوربین و geometry scene نیز critical configu- ration می‌سازد. pipeline حرفه‌ای باید:

۱. sample minimal را پیش از solve از نظر rank/area آزمون کند؛
۲. چند مدل محتمل مانند H و F را مقایسه کند؛
۳. penalty complexity یا prior geometric داشته باشد؛
۴. cheirality و محدودیت‌های دوربین را پس از بازیابی pose بررسی کند.

معیار تصمیم برای کاربرد

برای panorama صفحه‌ای یا دوران خالص، homography مناسب است؛ برای دو دید عمومی صحنه سه‌بعدی، fundamental/essential؛ برای localization با نقاط سه‌بعدی، PnP؛ و برای جسم non-rigid یک مدل سراسری واحد کافی نیست. انتخاب مدل پیش از RANSAC بخشی از فرض مسئله است.

۹.۹ از match به بازشناسی: سه خانواده کلاسیک

بازشناسی را می‌توان به سه صورت دید:

۱. **recognition: instance** آیا همین شیء/نما قبلاً دیده شده است؟
۲. **recognition: category** آیا تصویر به کلاس مفهومی تعلق دارد؟
۳. **retrieval:** تصاویر مشابه را با ranking برگردان.

ویژگی محلی برای instance و geometry بسیار طبیعی است؛ recognition category به تجمیع و classfiier نیاز دارد.

۱.۹.۹ تطبیق مستقیم قالب، chamfer و Hough generalized

برای edge های template map T و تصویر، distance chamfer با distance transform D_I:

$$E_{\text{chamfer}}(T, I) = \frac{۱}{|T|} \sum_{\boldsymbol{x} \in T} D_I(\boldsymbol{x}) \tag{۵۲.۹}$$

است. نسخه جهت‌دار اختلاف tangent را نیز جریمه می‌کند. Hough Generalized برای هر نقطه مرزی، بردار به مرکز مرجع را در R-table ذخیره و در تصویر جدید در فضای وضعیت رأی‌گیری می‌کند. این روش برای silhouette مشخص و تعداد کلاس کم تفسیرپذیر است، اما فضای رأی با scale/rotation بزرگ می‌شود.

۲.۹.۹ ویژگی‌های کلی شکل و ممان

ممان‌های هندسی

$$m_{pq} = \sum_x \sum_y x^p y^q I(x, y) \tag{۵۳.۹}$$

و ممان مرکزی

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q I(x, y) \tag{۵۴.۹}$$

برای مرکز، جهت و invariants استفاده می‌شوند. Hu هفت ترکیب ناوردا نسبت به انتقال، scale و rotation ساخت. این توصیفگرها compact و مناسب شکل تمیزند، ولی به error segmentation و clutter حساس‌اند.

۳.۹.۹ PCA و Eigenfaces

بردار تصویر $\boldsymbol{x} \in \mathbb{R}^D$ حول میانگین مرکز می‌شود. PCA جهت‌های بیشینه واریانس را از eigenvectors کوواریانس

$$\boldsymbol{S}_T = \frac{۱}{N} \sum_{i=۱}^{N} (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^\top \tag{۵۵.۹}$$

می‌یابد:

$$\boldsymbol{W}_{\text{PCA}} = \arg\max_{\boldsymbol{W}^\top \boldsymbol{W} = I} \operatorname{tr}(\boldsymbol{W}^\top \boldsymbol{S}_T \boldsymbol{W}). \tag{۵۶.۹}$$

Eigenfaces تصویر را در subspace کم‌بعد بازنمایی و neighbor nearest را ممکن می‌کند [۲۲]. PCA label را نمی‌بیند؛ تغییر نور ممکن است از تفاوت هویت واریانس بیشتری داشته باشد.

۴.۹.۹ LDA و Fisherfaces

با میانگین کلاس μ_c:

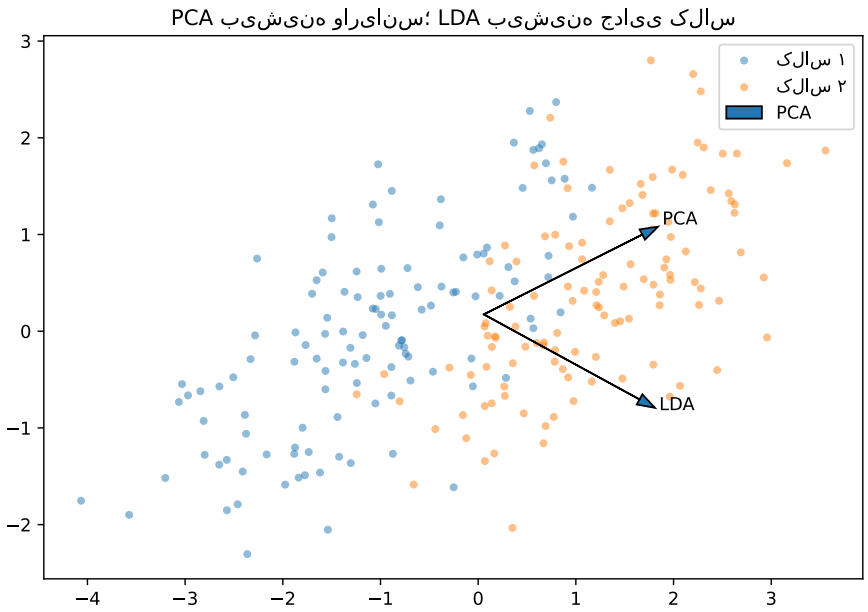
$$\boldsymbol{S}_W = \sum_c \sum_{i:y_i=c} (\boldsymbol{x}_i - \boldsymbol{\mu}_c)(\boldsymbol{x}_i - \boldsymbol{\mu}_c)^\top, \tag{۵۷.۹}$$

$$\boldsymbol{S}_B = \sum_c N_c(\boldsymbol{\mu}_c - \boldsymbol{\mu})(\boldsymbol{\mu}_c - \boldsymbol{\mu})^\top. \tag{۵۸.۹}$$

LDA نسبت جدایی بین کلاس به درون کلاس را بیشینه می‌کند:

$$\boldsymbol{W}_{\text{LDA}} = \arg\max_{\boldsymbol{W}} \frac{|\boldsymbol{W}^\top \boldsymbol{S}_B \boldsymbol{W}|}{|\boldsymbol{W}^\top \boldsymbol{S}_W \boldsymbol{W}|}. \tag{۵۹.۹}$$

راه‌حل از مسئله generalized eigenvalue $S_B\boldsymbol{w} = \lambda S_W \boldsymbol{w}$ می‌آید. در $D \gg N$، S_W تکین است و ابتدا PCA یا regularization لازم است [۲۳].



شکل ۸.۹: PCA جهت واریانس کل را می‌گیرد؛ LDA جهت جدایی برچسب‌دار را. هر دو خطی و وابسته به preprocessing و split صحیح‌اند.

۱۰.۹ طبقه‌بندهای کلاسیک روی ویژگی

۹.۱.۱ kNN و هندسه فضای ویژگی

قاعده نزدیک‌ترین همسایه برای نمونه $\boldsymbol{x}$:

$$\widehat{y} = \mathrm{mode}\{y_i : i \in \mathcal{N}_k(\boldsymbol{x})\} \tag{۹.۶۰}$$

است. ساده و بدون آموزش پارامتری است، اما هزینه inference با اندازه داده رشد می‌کند و به scaling ابعاد و dimensionality of curse حساس است. نسخه وزن‌دار از $w_i = (d_i + \varepsilon)^{-p}$ استفاده می‌کند. انتخاب k باید داخل validation و پس از تمام مراحل normalization feature انجام شود.

۹.۱.۲ مدل مولد گاوسی: LDA و QDA به‌عنوان classfiier

اگر

$$p(\boldsymbol{x} \mid y = c) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c), \tag{۹.۶۱}$$

آنگاه log-posterior تا ثابت برابر

$$\delta_c(\boldsymbol{x}) = -\frac{۱}{۲}\log|\boldsymbol{\Sigma}_c| - \frac{۱}{۲}(\boldsymbol{x} - \boldsymbol{\mu}_c)^{\top}\boldsymbol{\Sigma}_c^{-۱}(\boldsymbol{x} - \boldsymbol{\mu}_c) + \log \pi_c. \tag{۹.۶۲}$$

است. covariance مشترک مرز خطی LDA و covariance مستقل مرز درجه دوم QDA می‌دهد. shrinkage

$$\widehat{\Sigma}_\alpha = (۱ - \alpha)\widehat{\Sigma} + \alpha\frac{\mathrm{tr}(\widehat{\Sigma})}{D}\boldsymbol{I} \tag{۹.۶۳}$$

برای داده کم و feature پربعد ضروری است.

۹.۱.۳ ماشین بردار پشتیبان

برای داده دودویی $(\boldsymbol{x}_i, y_i)$، SVM: soft-margin

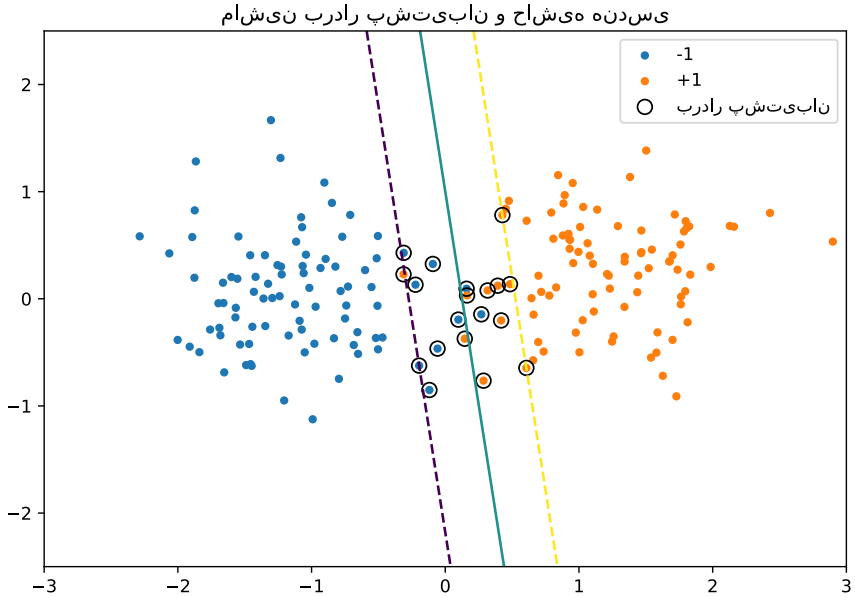
$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{۱}{۲}\|\boldsymbol{w}\|_۲^۲ + C\sum_i \xi_i \quad \text{s.t.} \quad y_i(\boldsymbol{w}^{\top}\boldsymbol{x}_i + b) \geq ۱ - \xi_i, \quad \xi_i \geq ۰. \tag{۹.۶۴}$$

فاصله مرز تا نزدیک‌ترین نمونه‌ها $۲/\|\boldsymbol{w}\|$ است؛ کمینه‌کردن نرم، حاشیه را بیشینه می‌کند. دوگان:

$$\max_{\boldsymbol{\alpha}} \sum_i \alpha_i - \frac{۱}{۲}\sum_{i,j}\alpha_i\alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j), \quad ۰ \leq \alpha_i \leq C, \quad \sum_i \alpha_i y_i = ۰. \tag{۹.۶۵}$$

تنها نمونه‌های $\alpha_i > ۰$ بردار پشتیبان‌اند. برای histogramها، kernelهای $\chi^۲$ یا intersection می‌توانند مناسب‌تر از RBF باشند، اما SVM linear روی HOG، VLAD و FV پس از نرمال‌سازی اغلب baseline

قوی و سریع است [۲۴، ۱۶].



شکل ۹.۹: SVM خطی. بردارهای پشتیبان مرز را تعیین می‌کنند؛ نقاط دورتر در جواب primal اثر مستقیم ندارند.

۹.۱۰.۴ چندکلاسه و calibration

راهبردهای one-vs-rest و one-vs-one چند classfiier دودویی می‌سازند. score حاشیه احتمال نیست. Platt scaling مدل logistic

$$P(y = ۱ \mid f) = \frac{۱}{۱ + \exp(Af + B)} \tag{۹.۶۶}$$

را روی validation جدا برازش می‌کند؛ isotonic regression انعطاف‌پذیرتر اما پرریسک‌تر است. cali-bration set نباید با test یکی باشد.

۹.۱۰.۵ AdaBoost و cascade

برای weak های learner $h_t(\boldsymbol{x}) \in \{-۱, +۱\}$، AdaBoost classfiier

$$F_T(\boldsymbol{x}) = \sum_{t=۱}^{T} \alpha_t h_t(\boldsymbol{x}) \tag{۹.۶۷}$$

و زیان نمایی

$$\sum_i \exp\left(-y_i F_T(\boldsymbol{x}_i)\right) \tag{۹.۶۸}$$

را به‌صورت stagewise کاهش می‌دهد. وزن نمونه‌ها

$$w_i^{(t+1)} \propto w_i^{(t)} \exp(-\alpha_t y_i h_t(\boldsymbol{x}_i)) \tag{۶۹.۹}$$

نمونه‌های اشتباه را تقویت می‌کند. Viola–Jones از Haar-like های feature سریع با image integral و cascade استفاده کرد: های stage نخست تقریباً همه پنجره‌های منفی آسان را با هزینه کم رد می‌کنند، های stage بعدی دقیق‌ترند [۲۵].

منطق cascade

اگر stage s نرخ detection d_s و positive false f_s داشته باشد، برای stage: S

$$D = \prod_{s=1}^{S} d_s, \qquad F = \prod_{s=1}^{S} f_s.$$

کاهش شدید positive false نیازمند حفظ detection بسیار بالا در هر stage است. cascade یک طراحی آماری هزینه‌محور است، نه صرفاً چند classfiier پشت‌سرهم.

۱۱.۹ بازنمایی کلمات بصری: از ویژگی محلی به بردار تصویر

ویژگی‌های محلی تعداد متغیر و ترتیب نامشخص دارند. Words Visual of Bag آن‌ها را به histogram با طول ثابت تبدیل می‌کند.

۱.۱۱.۹ ساخت واژگان

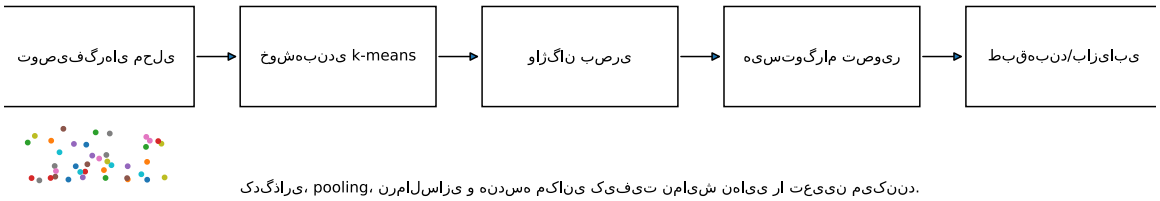
از های descriptor مجموعه train، نمونه‌برداری و k-means حل می‌شود:

$$\min_{\{\boldsymbol{c}_k\},\{a_i\}} \sum_i \|\boldsymbol{d}_i - \boldsymbol{c}_{a_i}\|_2^2 . \tag{۷۰.۹}$$

هاcentroid واژه بصری‌اند. هر descriptor به نزدیک‌ترین واژه quantize و histogram

$$h_k(I) = \sum_{i \in I} \mathbb{1}[a_i = k] \tag{۷۱.۹}$$

ساخته می‌شود. واژگان باید فقط روی train ساخته شود؛ ساخت codebook روی کل leakage dataset است.



شکل ۹.۱۰: خط لوله BoVW. انتخاب descriptor، codebook، weighting assignment و pooling همگی بخشی از مدل‌اند.

۹.۱۱.۲ Hard، soft و multiple assignment

Hard assignment discontinuous است. در soft assignment:

$$w_{ik} = \frac{\exp(-\beta \|\boldsymbol{d}_i - \boldsymbol{c}_k\|^2)}{\sum_\ell \exp(-\beta \|\boldsymbol{d}_i - \boldsymbol{c}_\ell\|^2)}, \qquad h_k = \sum_i w_{ik}. \tag{۹.۷۲}$$

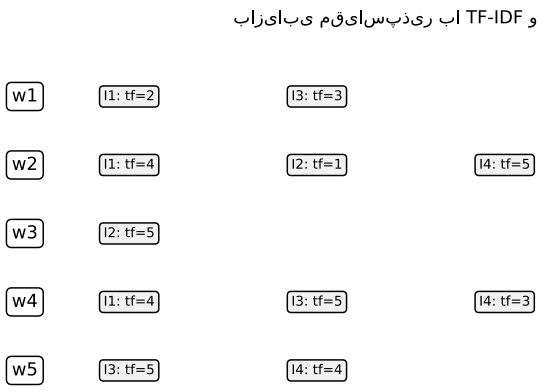
multiple assignment فقط چند centroid نزدیک را فعال می‌کند. soft quantization assignment error را کم، اما sparsity و سرعت inverted index را کاهش می‌دهد.

۹.۱۱.۳ TF–IDF و فهرست معکوس

برای تصویر j و واژه k:

$$\mathrm{tf}_{kj} = \frac{n_{kj}}{\sum_\ell n_{\ell j}}, \qquad \mathrm{idf}_k = \log \frac{N+1}{N_k+1}, \qquad v_{kj} = \mathrm{tf}_{kj}\mathrm{idf}_k. \tag{۹.۷۳}$$

واژه‌ای که در همه تصاویر است قدرت تشخیص کمی دارد. inverted index برای هر واژه posting list تصاویر دارای آن را ذخیره و جست‌وجو را به اشتراک واژه‌ها محدود می‌کند؛ این ایده Google video و retrieval image کلاسیک را مقیاس‌پذیر کرد [۲۷، ۲۸].



شکل ۱۱.۹: فهرست معکوس. واژه‌های query به posting listها می‌روند؛ TF–IDF و normalization score تصویر را می‌سازند.

۴.۱۱.۹ burstiness و stop words

بافت تکراری ممکن است یک واژه را ده‌ها بار در یک تصویر تکرار کند و score را مصنوعی بالا ببرد. راهکارها:

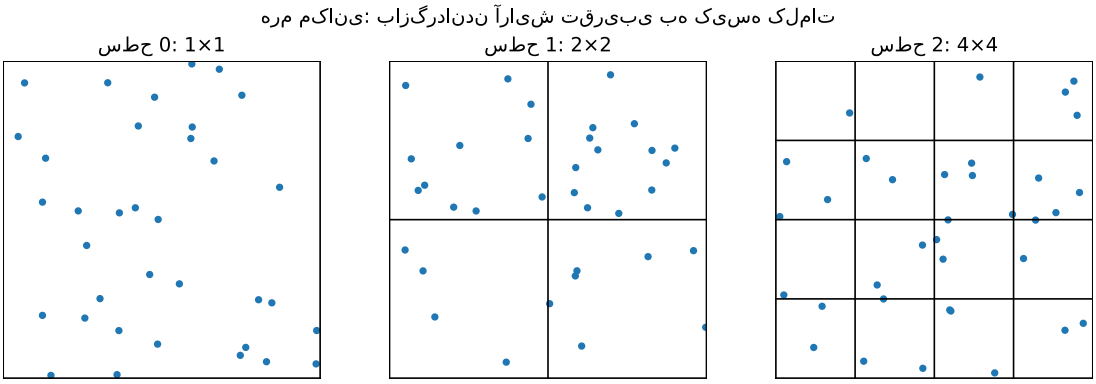
- normalization power $h_i \leftarrow \mathrm{sign}(h_i)|h_i|^{\alpha}$ با $0 < \alpha < 1$؛
- term capped، frequency؛
- حذف stop visual های word بسیار رایج؛
- verfiication geometric پس از retrieval اولیه.

۵.۱۱.۹ Matching Pyramid Spatial

BoVW ترتیب مکانی را حذف می‌کند. Pyramid Spatial تصویر را در سطح ℓ به $2^{\ell} \times 2^{\ell}$ سلول تقسیم و های histogram سلولی را concat می‌کند:

$$\Phi(I) = \left[\sqrt{w_0}\boldsymbol{h}^{(0)}, \sqrt{w_1}\boldsymbol{h}^{(1)}, \ldots, \sqrt{w_L}\boldsymbol{h}^{(L)}\right]. \tag{۷۴.۹}$$

intersection histogram kernel، weighted تطابق ریزتر را بیشتر وزن می‌دهد [۲۹].



شکل ۱۲.۹: هرم مکانی. نمایش به‌تدریج از orderless به حفظ آرایش تقریبی می‌رود، اما ابعاد با تعداد سلول‌ها رشد می‌کند.

الگوریتم ۵.۹ ـ آموزش BoVW بدون leakage

۱. split را در سطح شیء/ویدئو/فرد/صحنه انجام دهید.
۲. descriptorها را فقط از train و با سقف نمونه در هر تصویر جمع کنید.
۳. preprocessing مانند RootSIFT/PCA را فقط روی train برازش کنید.
۴. k-means را با چند initialization آموزش و distortion را ثبت کنید.
۵. train/validation/test را با codebook ثابت encode کنید.
۶. weighting و normalization را داخل pipeline اعمال کنید.
۷. classfiier و hyperparameterها را روی validation انتخاب کنید.
۸. test را یک بار ارزیابی و hash codebook و seed را ذخیره کنید.

۶.۱۱.۹ VLAD: تجمیع residual

BoVW فقط count را نگه می‌دارد. VLAD برای هر residual centroid های descriptor تخصیص‌یافته را جمع می‌کند:

$$\boldsymbol{v}_k = \sum_{i:a_i=k} (\boldsymbol{d}_i - \boldsymbol{c}_k), \qquad \boldsymbol{v} = [\boldsymbol{v}_1; \ldots; \boldsymbol{v}_K]. \tag{۷۵.۹}$$

intra-normalization هر block و سپس power+ ℓ_2 burstiness normalization را کنترل می‌کند. VLAD اطلاعات مرتبه اول نسبت به codeword را حفظ می‌کند و برای retrieval/recognition قوی‌تر از count خام است [۳۰].

۷.۱۱.۹ Vector: Fisher گرادیان log-likelihood

فرض کنید descriptorها از GMM با پارامتر $\lambda = \{\pi_k, \mu_k, \sigma_k\}$ آمده‌اند. Vector Fisher گرادیان log-likelihood داده نسبت به پارامتر مدل است:

$$\mathcal{G}_\lambda(I) = \frac{1}{N}\nabla_\lambda \log p(\{\boldsymbol{d}_i\} \mid \lambda). \tag{۷۶.۹}$$

برای covariance قطری و responsibility $\gamma_i(k)$، مؤلفه میانگین:

$$\boldsymbol{u}_k = \frac{1}{N\sqrt{\pi_k}} \sum_i \gamma_i(k) \frac{\boldsymbol{d}_i - \boldsymbol{\mu}_k}{\boldsymbol{\sigma}_k}, \tag{۷۷.۹}$$

و مؤلفه واریانس:

$$\boldsymbol{v}_k = \frac{1}{N\sqrt{2\pi_k}} \sum_i \gamma_i(k) \left[\frac{(\boldsymbol{d}_i - \boldsymbol{\mu}_k)^2}{\boldsymbol{\sigma}_k^2} - 1 \right]. \tag{۷۸.۹}$$

FV آمار مرتبه اول و دوم را نسبت به مدل مولد ثبت می‌کند. normalization power و SVM linear بخش جدایی‌ناپذیر اندpipeline [۳۱، ۳۲].

نمایش	اطلاعات نگه‌داشته	بعد تقریبی	مزیت	محدودیت
BoVW	count واژه	K	inverted sparse index،	quantization و حذف geometry
SPM	count در سلول	$K \sum 4^\ell$	ساختار مکانی	بعد زیاد، grid سخت
VLAD	residual مرتبه اول	KD	compact و قوی	dense و حساس به normalization
FV	آمار مرتبه ۱ و ۲ GMM	$2KD$	بیان غنی، linear classfiier	حافظه و پیچیدگی بیشتر

۱۲.۹ بازشناسی کلاسیک شیء و چهره: خط لوله کامل

۱.۱۲.۹ SVM + HOG برای آشکارسازی شیء

در detector لغزان، برای هر window بردار HOG ساخته و score خطی

$$s(\boldsymbol{x}) = \boldsymbol{w}^\top \Phi_{\text{HOG}}(\boldsymbol{x}) + b \tag{۷۹.۹}$$

محاسبه می‌شود. pyramid تصویر های scale شیء را پوشش می‌دهد. پنجره‌های هم‌پوشان با non-maximum suppression ادغام می‌شوند. در NMS کلاسیک، box با score بالاتر انتخاب و های box با

$$\mathrm{IoU}(B_i, B_j) = \frac{|B_i \cap B_j|}{|B_i \cup B_j|} > \tau_{\mathrm{nms}} \tag{۸۰۰۹}$$

حذف می‌شوند. mining hard-negative نقش مهمی دارد: پس از آموزش اولیه، false های positive واقعی detector از تصاویر منفی جمع و مدل دوباره آموزش داده می‌شود.

الگوریتم ۶.۹ ـ HOG–SVM با mining hard-negative

۱. نمونه‌های مثبت را هم‌تراز و window استاندارد تعریف کنید.
۲. از تصاویر منفی های window تصادفی اولیه بگیرید.
۳. HOG را با cell، bin block و normalization ثابت استخراج کنید.
۴. SVM linear را آموزش دهید.
۵. detector را روی تصاویر منفی کامل اجرا و false های positive پرامتیاز را جمع کنید.
۶. مجموعه منفی را با hard های negative گسترش و SVM را بازآموزی کنید.
۷. روی score threshold validation و NMS را انتخاب کنید؛ test را دست‌نخورده نگه دارید.

۲.۱۲.۹ Model Part Deformable

DPM شیء را با filter root و part های filter جابه‌جاشونده مدل می‌کند:

$$S(\boldsymbol{x}, \boldsymbol{z}) = \sum_i \boldsymbol{w}_i^{\mathsf{T}} \Phi(I, \boldsymbol{z}_i) - \sum_i \boldsymbol{d}_i^{\mathsf{T}} \psi(\Delta \boldsymbol{z}_i) + b. \tag{۸۱۰۹}$$

جمله deformation معمولاً شامل $[dx, dy, dx^۲, dy^۲]$ است. inference با transform distance سریع و آموزش با latent SVM انجام می‌شود. اهمیت تاریخی DPM در «appearance formalization + geometry + latent parts» است؛ ایده‌ای که در مدل‌های مدرن نیز بازمی‌گردد [۲۶].

۳.۱۲.۹ Eigenfaces، Fisherfaces و زیرفضاهای چهره

برای چهره، alignment از خود classfiier مهم‌تر است. crop، landmark scale و نور باید قرارداد ثابتی داشته باشند. pipeline کلاسیک:

$$\text{detect} \to \text{align} \to \text{normalize photometric} \to \text{PCA/LDA} \to \text{distance/classfiier}.$$

در open-set، threshold recognition فاصله برای رد هویت ناشناخته لازم است. closed-set accuracy بدون گزارش false reject accept/false برای کاربرد امنیتی کافی نیست.

۴.۱۲.۹ Histogram Pattern Binary Local

LBP برای پیکسل مرکز g_c و همسایه‌ها g_p:

$$\mathrm{LBP}_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p, \qquad s(z) = 1[z \geq 0]. \tag{۸۲.۹}$$

LBP Histogram در grid مکانی برای texture و چهره استفاده می‌شود. چون فقط ترتیب intensity را می‌گیرد، نسبت به تغییر monotonic روشنایی مقاوم است؛ اما نویز نزدیک threshold بیت‌ها را ناپایدار می‌کند. الگوهای unfiorm تعداد هایtransition ۱/۰ کم دارند و ابعاد را کاهش می‌دهند.

۱۳.۹ یادگیری متریک و بازشناسی مبتنی بر فاصله

گاهی classfiier ثابت کافی نیست و می‌خواهیم فاصله، شباهت معنایی یا هویتی را بازتاب دهد. Mahalanobis metric

$$d_M^2(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i - \boldsymbol{x}_j)^\top \boldsymbol{M}(\boldsymbol{x}_i - \boldsymbol{x}_j), \qquad \boldsymbol{M} \succeq 0 \tag{۸۳.۹}$$

با $\boldsymbol{M} = \boldsymbol{L}^\top \boldsymbol{L}$ معادل Euclidean بعد از projection است. یک objective ساده pairwise:

$$\min_{\boldsymbol{M} \succeq 0} \sum_{(i,j)\in\mathcal{S}} d_M^2(\boldsymbol{x}_i, \boldsymbol{x}_j) + \lambda \sum_{(i,j)\in\mathcal{D}} [m - d_M^2(\boldsymbol{x}_i, \boldsymbol{x}_j)]_+. \tag{۸۴.۹}$$

LMNN همسایه‌های هدف را نزدیک و هاimpostor را بیرون حاشیه نگه می‌دارد. این بخش پل مفهومی به loss contrastive/triplet است، اما در خط لوله کلاسیک می‌توان M را با optimization محدب آموخت.

۱۴.۹ تطبیق چندتصویری، مسیر و مکان

ویژگی و تطبیق تنها برای دو تصویر نیستند. در panorama، SfM SLAM و visual، graph localization مشاهده ساخته می‌شود:

$$\mathcal{G} = (\mathcal{V}_{\text{images}} \cup \mathcal{V}_{\text{tracks}}, \mathcal{E}_{\text{obs}}). \tag{۸۵.۹}$$

یک track مجموعه هایkeypoint است که یک نقطه سه‌بعدی را در چند تصویر می‌بینند. union کور pairwise هاmatch می‌تواند inconsistency بسازد؛ consistency cycle باید بررسی شود. اگر $i \leftrightarrow j$ و

$j \leftrightarrow k$، correspondence $i \leftrightarrow k$ باید با geometry سازگار باشد.

۱.۱۴.۹ retrieval سپس localization

برای پایگاه بزرگ، ابتدا descriptor global یا BoVW چند تصویر نامزد را بازیابی می‌کند؛ سپس local matching و PnP روی هر نامزد اجرا می‌شود. این طراحی coarse-to-fine هزینه را کاهش می‌دهد:

$$\text{retrieval global} \rightarrow \text{matching local} \rightarrow \text{PnP D۳D-۲} \rightarrow \text{refinement pose.}$$

در سیستم حرفه‌ای، retrieval recall@K و accuracy pose جدا گزارش می‌شوند تا معلوم شود شکست از بازیابی است یا هندسه.

۱.۵.۹ روش‌های آموختنی: ادامه یا جایگزین خط لوله کلاسیک؟

روش‌های آموختنی سه الگو دارند:

۱. detector و descriptor را می‌آموزند، اما matching و RANSAC کلاسیک می‌ماند؛
۲. context matcher، assignment را می‌آموزد، ولی keypoint خارجی می‌گیرد؛
۳. detector-free یا dense correspondence matching را مستقیماً از دو تصویر می‌سازد.

۱.۱.۵.۹ SuperPoint: آشکارسازی و توصیف sefl-supervised

SuperPoint یک encoder مشترک و دو detector/descriptor head دارد. ابتدا detector مصنوعی روی شکل‌های هندسی آموزش و با Adaptation Homographic روی تصاویر واقعی sefl-label تولید می‌شود. descriptor با هایcorrespondence ناشی از homography آموزش می‌بیند [۳۳]. مزیت آن detector co-design و descriptor است؛ محدودیت آن وابستگی به distribution آموزش و مدل sefl-supervision transformation است.

۲.۱.۵.۹ SuperGlue و LightGlue

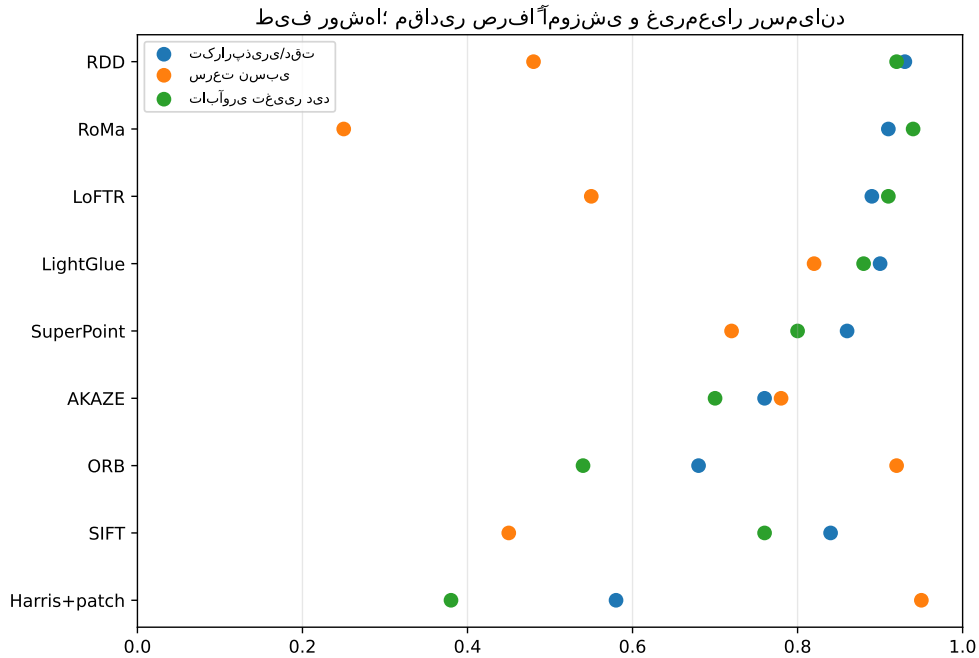
SuperGlue مجموعه هاkeypoint و هاdescriptor را با contextualize sefl/cross-attention و assign-ment را با dffierentiable transport optimal حل می‌کند [۳۶]. LightGlue طراحی سبک‌تر با stopping early confidence، و pruning point ارائه می‌کند؛ برای هایpair آسان عمق کمتری مصرف می‌کند و sparse matching را کاراتر می‌سازد [۳۸].

۳.۱۵.۹ LoFTR و matching detector-free

LoFTR به‌جای آشکارسازی مستقل، feature های map دو تصویر را با transformer مرتبط و coarse-to-fine match می‌کند. این روش در ناحیه کم‌بافت یا تکرارپذیری ضعیف keypoint مزیت دارد، اما های correspondence متراکم‌تر، حافظه و filtering متفاوت می‌طلبند [۳۷]. LoFTR Efficient در ۲۰۲۴ هدف کاهش هزینه و نزدیک‌شدن سرعت به روش sparse را دنبال کرد [۳۹].

۴.۱۵.۹ matching Dense و ویژگی‌های بنیادی

RoMa از های feature pretrained ۲DINOv برای robust matching dense بهره می‌گیرد و confidence/warp را پیش‌بینی می‌کند [۴۰]. OmniGlue ترکیب های feature foundation با keypoint matcher را برای generalization مطرح می‌کند [۴۱]. RDD در CVPR ۲۰۲۵ detector و descriptor مبتنی بر transformer deformable را برای تغییر دید و مقیاس شدید ارائه کرد [۴۲]. در CVPR ۲۰۲۶ نیز تطبیق گرهای مبتنی بر state-space و correlation sparse به مقیاس‌پذیری های pair بزرگ‌تر پرداخته‌اند [۴۳].



شکل ۱۳.۹: نمای آموزشی طیف کلاسیک تا آموختنی. نقاط شکل benchmark رسمی نیستند؛ هدف نمایش trade-off میان سرعت، robustness و complexity است.

پل به دهه آینده

دهه آینده احتمالاً «یک matcher واحد برای همه» نخواهد داشت. سیستم‌های واقعی از ترکیب زیر سود می‌برند:

- descriptor کلاسیک برای latency کم، CPU و explainability؛

- feature آموختنی برای تغییر، illumination viewpoint و texture-poor؛
- هندسه کلاسیک برای تضمین سازگاری فیزیکی؛
- uncertainty و detection failure برای جلوگیری از اعتماد کور؛
- foundation-based retrieval و local verfiication برای مقیاس بزرگ.

خواندن مقاله و استاندارد: مقایسه منصفانه کلاسیک و آموختنی

تعداد match خام را مقایسه نکنید. برای هر روش، detector، budget، resolution- preprocess ing، geometric matcher، residual threshold model و device compute را ثبت کنید. معیارها باید شامل precision، recall match، correspondence تعداد، pose inlier، AUC- fail ure، latency rate و memory باشند. استفاده RANSAC متفاوت می‌تواند نتیجه feature را وارونه کند.

۱۶.۹ ارزیابی آشکارساز و تطبیق گر

۱.۱۶.۹ repeatability

اگر N_1, N_2 تعداد نواحی قابل مشاهده و N_{rep} زوج تکرارشده باشد:

$$\text{repeatability} = \frac{N_{\text{rep}}}{\min(N_1, N_2)}. \tag{۸۶.۹}$$

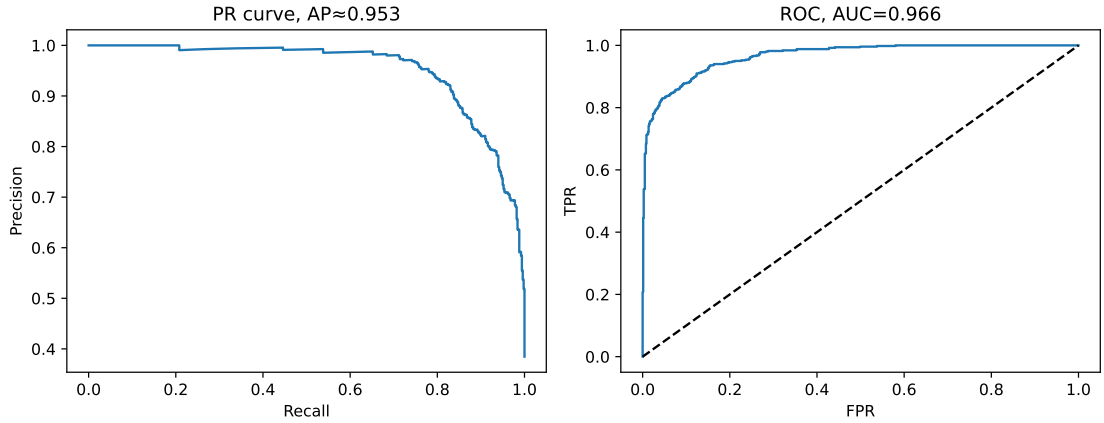
تعریف، visibility overlap threshold و correspondence یک‌به‌یک باید صریح باشد. repeatability بالا بدون matching discriminative descriptor خوب را تضمین نمی‌کند.

۲.۱۶.۹ matching، precision recall و score

اگر M match پذیرفته، C صحیح و G correspondence قابل مشاهده باشد:

$$\text{precision} = \frac{C}{M}, \qquad \text{recall} = \frac{C}{G}. \tag{۸۷.۹}$$

آستانه ratio یا confidence منحنی PR می‌سازد. area AP زیر PR است و در imbalance class مناسب‌تر از ROC می‌تواند باشد.



شکل ۱۴.۹: آستانه matcher نقطه operating را روی PR/ROC انتخاب می‌کند. یک عدد accuracy رفتار threshold را پنهان می‌کند.

۳.۱۶.۹ خطای هندسی و AUC pose

برای corner homography، reprojection؛ relative برای pose، خطای دوران و جهت انتقال:

$$e_R = \arccos\left(\frac{\operatorname{tr}(\boldsymbol{R}_{\text{gt}}\boldsymbol{R}_{\text{est}}^{\top}) - ۱}{۲}\right), \tag{۸۸.۹}$$

$$e_t = \arccos\left(\frac{\boldsymbol{t}_{\text{gt}}^{\top}\boldsymbol{t}_{\text{est}}}{\|\boldsymbol{t}_{\text{gt}}\|\,\|\boldsymbol{t}_{\text{est}}\|}\right). \tag{۸۹.۹}$$

معمولاً $e = \max(e_R, e_t)$ و AUC تا هایthreshold مانند ۵، ۱۰ و ۲۰ درجه گزارش می‌شود. median فقط شکست‌های کامل را پنهان می‌کند؛ rate failure نیز لازم است.

۴.۱۶.۹ ارزیابی بازشناسی

برای matrix confusion C_{ij}، average macro به هر کلاس وزن برابر و average micro به هر نمونه وزن برابر می‌دهد. در عدم توازن، accuracy ممکن است گمراه‌کننده باشد. برای retrieval:

$$\mathrm{AP}(q) = \frac{۱}{N_q^{+}}\sum_{k=۱}^{N} P_q(k)\,\mathrm{rel}_q(k), \qquad \mathrm{mAP} = \frac{۱}{Q}\sum_{q}\mathrm{AP}(q). \tag{۹۰.۹}$$

در retrieval instance باید، هاdistractor query، multi-scale crop و geometric re-ranking تعریف شوند.

۵.۱۶.۹ پروتکل robustness

تغییرات را جدا و ترکیبی بسنجید:

- blur، noise، gamma JPEG، و exposure؛

- rotation، scale affine viewpoint و perspective؛
- seasonal/day-night modality، RGB–IR underwater یا medical؛
- dynamic occlusion objects و repetitive texture؛
- rolling shutter و motion blur؛
- کمبود texture و saturation.

گزارش average روی همه corruptionها کافی نیست؛ failure taxonomy و بدترین گروه کاربردی مهم‌اند.

لایه ارزیابی	معیار	نکته گزارش
آشکارسازی	repeatability، localization، coverage	budget و visibility ثابت
توصیف/تطبیق هندسه	PR، AP matching score، inlier ratio، reprojection، pose AUC	metric و threshold ratio ثابت RANSAC و مدل یکسان
بازیابی	Recall@K، mAP query، time	scale distractor و index memory
بازشناسی	macro-F۱، balanced accuracy، ROC/PR	split گروهی و calibration
مهندسی	latency، throughput mem-ory، energy، CPU/GPU، batch=۱ و warm-up	
پایداری	corruption/OOD/failure rate	گروه‌بندی و فاصله اطمینان

۱۷.۹ آزمایشگاه کدنویسی فصل

۱.۱۷.۹ Harris از صفر با قرارداد نوع داده

```python
from __future__ import annotations
import numpy as np
from scipy.ndimage import gaussian_filter, maximum_filter

def harris_response(
    image: np.ndarray,
    sigma_d: float = 1.0,
    sigma_i: float = 2.0,
    k: float = 0.04,
) -> np.ndarray:
    """Compute Harris response on a 2-D image scaled to [0, 1]."""
```

```
    if image.ndim != 2:
        raise ValueError("image must be grayscale")
    x = image.astype(np.float64)
    if not np.isfinite(x).all():
        raise ValueError("image contains NaN/Inf")
    x = (x - x.min()) / (x.max() - x.min() + 1e-12)
    smooth = gaussian_filter(x, sigma_d)
    iy, ix = np.gradient(smooth)
    sxx = gaussian_filter(ix * ix, sigma_i)
    syy = gaussian_filter(iy * iy, sigma_i)
    sxy = gaussian_filter(ix * iy, sigma_i)
    det = sxx * syy - sxy * sxy
    trace = sxx + syy
    return det - k * trace * trace

def select_keypoints(response: np.ndarray, n: int = 1000, radius: int
    = 4):
    local_max = response == maximum_filter(response, size=2 * radius
   + 1)
    values = response[local_max]
    coords = np.argwhere(local_max)
    keep = np.argsort(values)[::-1][:n]
    return coords[keep], values[keep]
```

آزمایش: property-based با افزودن ثابت روشنایی پاسخ gradient تغییر نکند؛ با ضرب intensity پاسخ Harris به توان چهار scale شود؛ مختصات تحت translation داخلی همان translation را دنبال کند.

۲.۱۷.۹ RANSAC خط و تعمیم به homography

```
from dataclasses import dataclass
import numpy as np

@dataclass
class RansacResult:
    model: tuple[float, float]
    inliers: np.ndarray
    residuals: np.ndarray

def ransac_line(xy: np.ndarray, threshold: float, max_trials: int =
   2000,
                confidence: float = 0.999, seed: int = 0) ->
   RansacResult:
```

```
    rng = np.random.default_rng(seed)
    n = len(xy)
    if n < 2:
        raise ValueError("at least two points are required")
    best_mask = np.zeros(n, dtype=bool)
    best_model = (0.0, 0.0)
    trials = 0
    target_trials = max_trials
    while trials < min(max_trials, target_trials):
        i, j = rng.choice(n, size=2, replace=False)
        dx = xy[j, 0] - xy[i, 0]
        if abs(dx) < 1e-12:
            trials += 1
            continue
        a = (xy[j, 1] - xy[i, 1]) / dx
        b = xy[i, 1] - a * xy[i, 0]
        residuals = np.abs(a * xy[:, 0] - xy[:, 1] + b) / np.sqrt(a*a + 1)
        mask = residuals < threshold
        if mask.sum() > best_mask.sum():
            best_mask = mask
            best_model = (a, b)
            w = max(mask.mean(), 1e-12)
            p_good = max(1.0 - w**2, 1e-12)
            target_trials = int(np.ceil(np.log(1-confidence) / np.log(p_good)))
        trials += 1
    a, b = np.polyfit(xy[best_mask, 0], xy[best_mask, 1], 1)
    residuals = np.abs(a * xy[:, 0] - xy[:, 1] + b) / np.sqrt(a*a + 1)
    return RansacResult((float(a), float(b)), residuals < threshold, residuals)
```

برای، solver minimal homography چهار correspondence نرمال‌شده DLT است؛ residual را transfer symmetric بگیرید و های‌sample با مساحت کم را پیش از solve رد کنید.

۳.۱۷.۹ SVM + BoVW در pipeline امن

```
from sklearn.cluster import MiniBatchKMeans
from sklearn.preprocessing import normalize
from sklearn.svm import LinearSVC
import numpy as np
```

```python

def fit_codebook(train_descriptors: list[np.ndarray], k: int, seed: int = 0):
    pool = np.concatenate([d for d in train_descriptors if d is not None], axis=0)
    km = MiniBatchKMeans(n_clusters=k, batch_size=4096, random_state=seed,
                         n_init=10, reassignment_ratio=0.01)
    km.fit(pool)
    return km

def encode_bovw(descriptors: list[np.ndarray], codebook) -> np.ndarray:
    features = np.zeros((len(descriptors), codebook.n_clusters), dtype=np.float64)
    for i, d in enumerate(descriptors):
        if d is None or len(d) == 0:
            continue
        words = codebook.predict(d)
        features[i] = np.bincount(words, minlength=codebook.n_clusters)
    features = np.sqrt(features)  # power normalization for counts
    return normalize(features, norm="l2")

# IMPORTANT: descriptors_test never participate in codebook fitting.
codebook = fit_codebook(descriptors_train, k=512, seed=42)
Xtr = encode_bovw(descriptors_train, codebook)
Xva = encode_bovw(descriptors_val, codebook)
model = LinearSVC(C=1.0, class_weight="balanced").fit(Xtr, y_train)
```

قرارداد میان ریاضیات و کد

هر آزمایش باید چهار جدول جدا بسازد: detector، descriptor/matching statistics، statistics geometry و task statistics. ادغام همه چیز در accuracy نهایی، محل شکست را پنهان می‌کند. همه arrayها، thresholdها و random seedها باید ذخیره شوند.

۱۸۰۹ طراحی آزمایش‌های عمیق و بازتولیدپذیر

۱.۱۸.۹ آزمایش ۱: ناوردایی کنترل‌شده

یک تصویر مرجع و grid transformation بسازید:

$$\theta \in [-۹۰^\circ, ۹۰^\circ], \quad s \in [۰٫۵, ۲], \quad \text{blur} \in [۰, ۴], \quad \gamma \in [۰٫۵, ۲], \quad \text{noise} \in [۰, ۰٫۱].$$

برای Harris+patch، SIFT ORB و AKAZE، repeatability precision match و homography error را رسم کنید. هر transformation را تنها و سپس دوبه‌دو اعمال کنید تا interaction دیده شود.

۲.۱۸.۹ آزمایش ۲: threshold و trade-off

برای ratio ρ از ۵٫۰ تا ۱، curve PR و runtime RANSAC را ثبت کنید. ratio سخت precision را بالا می‌برد، اما ممکن است inlier کافی برای model minimal باقی نگذارد. point operating باید با success pose، نه فقط descriptor، precision انتخاب شود.

۳.۱۸.۹ آزمایش ۳: budget descriptor

تعداد keypoint را ۲۵۰, ۵۰۰, ۱۰۰۰, ۲۰۰۰, ۴۰۰۰ تغییر دهید. زمان detect، describe match و memory RANSAC، AUC pose و را گزارش کنید. saturation curve نشان می‌دهد نقطه بیشتر چه زمانی فقط هزینه و ambiguity می‌افزاید.

۴.۱۸.۹ آزمایش ۴: BoVW، VLAD و FV

با descriptor ثابت، size codebook و dimension PCA را تغییر دهید. برای هر representation، power normalization، ℓ_2 و classfiier ثابت را مقایسه کنید. codebook train، feature time، disk dimension، inference size و macro-F۱/mAP گزارش شود.

۵.۱۸.۹ آزمایش ۵: کلاسیک در برابر آموختنی

SIFT+ratio+MAGSAC++، SuperPoint+LightGlue Efficient LoFTR و یک matcher dense را روی های subset، viewpoint illumination و texture-poor اجرا کنید. resolution و geometric verfiier را تا حد ممکن یکسان کنید. علاوه بر average، بدترین decile و rate failure را گزارش کنید.

۶.۱۸.۹ آزمایش ۶: shfit domain

مدل یا threshold را روی indoor/day تنظیم و روی outdoor/night، RGB–IR یا تصاویر صنعتی آزمون کنید. روش کلاسیک نیز hyperparameter دارد؛ ادعای «بدون آموزش» به معنی «بدون تنظیم» نیست.

۱۹.۹ خطاهای رایج در مقاله و پروژه

۱. نمایش چند خط match رنگی و نتیجه‌گیری کیفی بدون ground truth.
۲. گزارش تعداد match خام به‌جای inlier هندسی و pose accuracy.
۳. استفاده matcher یا metric ناسازگار با descriptor.
۴. تنظیم ratio/RANSAC روی test.
۵. ساخت PCA codebook، یا statistics normalization روی کل داده.
۶. split تصادفی های frame یک ویدئو در train و test.
۷. مقایسه SIFT در resolution کامل با مدل عمیق در resolution کوچک یا بالعکس.
۸. نادیده‌گرفتن extraction feature latency و فقط گزارش matcher.
۹. استفاده homography برای صحنه سه‌بعدی عمومی و تفسیر inlier زیاد به‌عنوان درستی pose.
۱۰. حذف های pair شکست‌خورده از میانگین.
۱۱. گزارش تنها accuracy در open-set recognition.
۱۲. استفاده تصویر نمونه مشهور و عدم ارزیابی روی dataset واقعی دامنه هدف.

۲۰.۹ پرسش‌های تصمیم مهندسی

پرسش	پاسخ طراحی
CPU و latency سخت؟	ORB/AKAZE، Hamming grid coverage و geometry-try کلاسیک را baseline کنید.
تغییر دید و نور شدید؟	SIFT/RootSIFT یا feature آموختنی؛ affine/dense matcher و verfiier robust.
بافت کم؟	detector-free/dense matching یا line/region feature؛ keypoint صرف ممکن است پوشش ندهد.
پایگاه میلیون‌تصویری؟	global/coarse retrieval، inverted index/PQ، سپس local verfiication.
نیاز به توضیح و certfiication؟	inlier residual، mask، uncertainty geometry و fall-back کلاسیک حفظ شود.
دامنه خاص و داده کم؟	strong classical baseline + adaptation سبک؛ از fine-tune بزرگ بدون validation پرهیز.
چهره/هویت open-set؟	threshold metric، rejection ROC/FAR/FRR و group-aware split.

قطعه صنعتی ثابت؟	template/edge/chamfer ممکن است از شبکه پیچیده مناسب‌تر و قابل ممیزی‌تر باشد.

۲۱۰۹ تمرین‌های نظری و اثباتی

۱. ناوردایی ZNCC نسبت به $I' = aI + b$ برای $a > ۰$ را اثبات کنید و حالت $a < ۰$ را تحلیل کنید.

۲. معادله تانسور ساختار (۱۴۰۹) را از تیلور استخراج کنید.

۳. نواحی تصمیم Harris را در صفحه $(\lambda_۱, \lambda_۲)$ برای چند k رسم کنید.

۴. نشان دهید مقیاس پاسخ Harris با ضرب intensity چگونه تغییر می‌کند.

۵. رابطه DoG و LoG را با بسط تیلور نسبت به σ استخراج کنید.

۶. برای blob Gaussian دوبعدی، مقیاس اکسترمم $\sigma^۲\nabla^۲ L$ را محاسبه کنید.

۷. معادله offset زیرپیکسلی (۲۵۰۹) را از شرط ایستایی به دست آورید.

۸. رابطه rejection edge در SIFT را برحسب نسبت eigenvalue اثبات کنید.

۹. نشان دهید برای بردارهای واحد، ranking فاصله Euclidean و cosine یکسان است.

۱۰. یک distribution بسازید که test ratio بهتر از threshold مطلق عمل کند و یک مثال عکس آن ارائه دهید.

۱۱. فرمول RANSAC iteration را استخراج و برای $w = ۰٫۳$ و مدل‌های ۲، ۴، ۵ و ۸ نقطه‌ای محاسبه کنید.

۱۲. اثر overestimated و underestimated RANSAC threshold را بر bias و variance تحلیل کنید.

۱۳. شرط تباهی چهار نقطه homography DLT را با matrix rank بیان کنید.

۱۴. دوگان SVM را از primal Lagrangian استخراج کنید.

۱۵. نشان دهید label-invariant PCA است و مثال دوکلاسی بسازید که مؤلفه اول برای طبقه‌بندی نامناسب باشد.

۱۶. generalized LDA eigenproblem را از quotient Rayleigh استخراج کنید.

۱۷. نشان دهید TF–IDF چگونه واژه موجود در همه تصاویر را کم‌وزن می‌کند.

۱۸. ابعاد SPM، VLAD و FV را برای $K = ۲۵۶, D = ۶۴, L = ۲$ محاسبه کنید.

۱۹. GMM log-likelihood gradient نسبت به μ_k را استخراج و فرمول mean FV را توجیه کنید.

۲۰. رابطه matcher precision/recall و RANSAC probability success را به‌صورت تحلیلی بحث کنید.

۲۲۰۹ پروژه‌های پایان فصل

۱۰۲۲۰۹ پروژه A: آزمایشگاه ویژگی کلاسیک

Harris، Shi–Tomasi، FAST DoG و Hessian را با API مشترک بسازید. repeatability، localiza- tion، coverage و runtime را روی های transformation کنترل‌شده مقایسه کنید.

۲۰۲۲۰۹ پروژه B: موتور تطبیق و panorama

SIFT/ORB/AKAZE، ratio، mutual، RANSAC/MAGSAC++ homography و blending را پیاده‌سازی کنید. failure ناشی از parallax، motion و repetition texture را طبقه‌بندی کنید.

۳۰۲۲۰۹ پروژه C: بازشناسی کلاسیک کامل

روی یک dataset چندکلاسه، HOG+SVM، LBP+SVM، BoVW، VLAD SPM و FV را با split گروهی مقایسه کنید. ابعاد، زمان، حافظه، macro-F۱ و calibration را گزارش کنید.

۴۰۲۲۰۹ پروژه D: بازیابی تصویر مقیاس‌پذیر

TF–IDF، inverted index، stop-word، burstiness و geometric re-ranking را بسازید. Re- call@K، mAP query latency و size index را در اندازه پایگاه افزاینده اندازه‌گیری کنید.

۵۰۲۲۰۹ پروژه E: کلاسیک در برابر LightGlue/LoFTR

روی HPatches یا یک مجموعه دامنه‌خاص، SIFT+MAGSAC++، SuperPoint+LightGlue و LoFTR را در پروتکل ثابت مقایسه کنید. نتیجه را به viewpoint، illumination و texture تقسیم کنید.

۶۰۲۲۰۹ پروژه F: open-set چهره یا شیء

PCA/LDA/metric را با gallery/probe مستقل بسازید. CMC، ROC، FAR/FRR و rejection ناشناخته را گزارش و leakage هویت را کنترل کنید.

۲۳۰۹ پرامپت پیشنهادی برای Codex

```
Build a fully reproducible Python research package for Chapter 9,
"Local Features, Matching, and Classical Recognition".

Implement from scratch where educationally useful:
```

```
- ZNCC and FFT phase correlation;
- Harris/Shi-Tomasi with subpixel localization and ANMS;
- Gaussian/DoG scale space, 3-D extrema, contrast and edge rejection;
- a compact educational SIFT-like descriptor with trilinear voting;
- BRIEF-style binary tests and Hamming matching;
- brute-force 2-NN, ratio test, mutual check and assignment diagnostics;
- adaptive RANSAC for line, homography and fundamental matrix, including
  degeneracy tests, normalized solvers and nonlinear refinement;
- detector repeatability, overlap error, matching PR/AP, reprojection error,
  relative-pose AUC and failure rate;
- BoVW, TF-IDF, inverted index, spatial pyramid, VLAD and Fisher Vector;
- PCA, LDA, kNN, linear/RBF SVM, calibrated scores and AdaBoost;
- HOG + sliding-window detection with hard-negative mining and NMS.

Use trusted library wrappers for SIFT, ORB, AKAZE, MAGSAC++, SuperPoint,
LightGlue, Efficient LoFTR, RoMa or other learned methods when licenses and
weights are available. Never silently download weights. Record checkpoint,
commit, license, image resolution, device and preprocessing.

Research protocol:
1. Use analytic synthetic transformations plus public benchmarks and one
   domain-specific dataset.
2. Split by object/scene/person/video before any PCA, codebook, whitening or
   hyperparameter fitting.
3. Compare methods at fixed keypoint budgets and fixed geometric verification.
4. Sweep ratio/confidence and RANSAC thresholds; produce PR and pose-success
   curves, not one cherry-picked operating point.
5. Log detect, describe, match, verify and end-to-end latency separately.
6. Save keypoints, descriptors metadata, matches, inlier masks, model matrices,
```

```
   residuals, seeds, configs, environment and data hashes.
7. Add robustness tests for scale, rotation, affine viewpoint, blur, noise,
   exposure, gamma, JPEG, occlusion, repeated texture and low texture.
8. Report paired bootstrap confidence intervals over independent image pairs
   or queries and preserve all failed pairs in the denominator.
9. Add unit/property tests for coordinate order, dtype/range, empty descriptors,
   Hamming vs L2 selection, normalized DLT, RANSAC degeneracy, deterministic
   sampling, label leakage and metric edge cases.
10. Provide a single command that regenerates every figure, CSV and table used
   by the chapter, plus an automatic failure gallery sorted by declared error.
```

۲۴۰۹ تقسیم پیشنهادی ویدئوهای ۱۵ دقیقه‌ای

بخش	موضوع
۱	مسئله ویژگی، تطبیق و بازشناسی
۲	هم‌وردایی و ناوردایی با گروه تبدیل
۳	معیارهای localization repeatability، و coverage
۴	SSD و مدل نویز گاوسی
۵	SAD، NCC و ZNCC
۶	تصویر انتگرالی و تطبیق سریع
۷	correlation phase و قضیه شیفت فوریه
۸	انرژی تغییر وصله و Moravec
۹	استخراج تانسور ساختار
۱۰	Harris و صفحه مقادیر ویژه
۱۱	Shi–Tomasi و ANMS
۱۲	FAST، AGAST و درخت تصمیم
۱۳	ضرورت فضای مقیاس
۱۴	فضای مقیاس گاوسی و axioms
۱۵	مشتق نرمال‌شده و LoG
۱۶	استخراج DoG از معادله گرما

۱۷ اکسترمم سه‌بعدی مکان-مقیاس
۱۸ مکان‌یابی زیرپیکسلی
۱۹ حذف کنتراست پایین و پاسخ لبه
۲۰ SURF Hessian، و blob
۲۱ KAZE/AKAZE و dffiusion غیرخطی
۲۲ MSER و ناحیه stable maximally
۲۳ region covariant affine
۲۴ جهت گذاری canonical
۲۵ descriptor وصله خام و whitening
۲۶ ساخت SIFT مرحله‌به‌مرحله
۲۷ interpolation و نرمال‌سازی SIFT
۲۸ RootSIFT و فاصله Hellinger
۲۹ HOG و normalization block
۳۰ Context Shape و تطبیق contour
۳۱ BRIEF و آزمون‌های دودویی
۳۲ ORB، BRISK FREAK و AKAZE
۳۳ فاصله L۲، Mahalanobis cosine، و Hamming
۳۴ neighbor nearest و dimensionality of curse
۳۵ test ratio و تحلیل آماری آن
۳۶ check mutual و assignment
۳۷ FLANN، LSH و quantization product
۳۸ residual هندسی و انتخاب مدل
۳۹ استخراج احتمال RANSAC iteration
۴۰ uncertainty threshold، و فاصله Mahalanobis
۴۱ PROSAC، USAC LO-RANSAC و MAGSAC++
۴۲ degeneracy و selection model
۴۳ Hough template/chamfer/generalized
۴۴ ممان‌ها و هایdescriptor کلی شکل
۴۵ PCA و Eigenfaces
۴۶ LDA و Fisherfaces
۴۷ kNN، LDA/QDA classfiier
۴۸ SVM، margin primal و dual
۴۹ kernel، چندکلاسه و calibration
۵۰ AdaBoost، Haar و cascade
۵۱ HOG+SVM و mining hard-negative
۵۲ DPM و parts latent
۵۳ Bag of Visual Words و k-means
۵۴ assignment hard/soft و quantization
۵۵ TF-IDF و فهرست معکوس

۵۶	burstiness و stop words
۵۷	Spatial Pyramid Matching
۵۸	VLAD و residual aggregation
۵۹	Vector Fisher و GMM gradient
۶۰	retrieval سپس geometric verfiication
۶۱	consistency cycle track، و چندتصویری
۶۲	SuperPoint و sefl-supervision هندسی
۶۳	SuperGlue و assignment آموختنی
۶۴	LightGlue و تطبیق تطبیقی سریع
۶۵	LoFTR و matching detector-free
۶۶	Efficient LoFTR، RoMa و OmniGlue
۶۷	RDD و مسیر پژوهش ۲۰۲۵–۲۰۲۶
۶۸	پروتکل repeatability و PR matching
۶۹	pose، AUC error و rate failure
۷۰	mAP، macro/micro confusion، و open-set
۷۱	robustness و shfit domain
۷۲	کارگاه Harris و scale-space
۷۳	کارگاه SIFT/ORB و RANSAC
۷۴	کارگاه BoVW/VLAD/FV و SVM
۷۵	طراحی مقاله و جلوگیری از leakage
۷۶	پروژه‌های پایان فصل و جمع‌بندی

۲۵.۹ جمع‌بندی و پل به فصل دهم

جمع‌بندی فصل

۱. ویژگی خوب باید مکان را هم‌وردا، توصیف را نسبت به تغییرات مجاز پایدار و میان ساختارهای متفاوت متمایز کند.

۲. matching template یک baseline آماری شفاف است؛ ZNCC تغییر آفین شدت و phase correlation انتقال را مدیریت می‌کند.

۳. Harris از انرژی تغییر وصله و های eigenvalue تانسور ساختار به دست می‌آید؛ FAST سرعت را با آزمون دایره و درخت تصمیم می‌خرد.

۴. فضای مقیاس مسئله اندازه ویژگی را حل می‌کند؛ DoG تقریب محاسباتی LoG و SIFT نمونه کلاسیک انتخاب مکان–مقیاس–جهت است.

۵. SIFT با pooling جهت و مکان به پایداری می‌رسد؛ RootSIFT histogram metric را

اصلاح و هایdescriptor دودویی سرعت و حافظه را کاهش می‌دهند.

۶. neighbor nearest بدون ،mutuality ratio و geometry تنها فرضیه خام است.

۷. RANSAC و جانشین‌های آن matching را به استدلال هندسی فیزیکی متصل می‌کنند؛ thresh-old و degeneracy جزء مدل‌اند.

۸. PCA، LDA، boosting SVM و HOG نشان می‌دهند بازشناسی کلاسیک ترکیب نمایش و تصمیم است، نه یک الگوریتم منفرد.

۹. BoVW ترتیب را حذف، Pyramid Spatial بخشی از آن را بازمی‌گرداند و VLAD/FV آمار غنی‌تری از هاresidual می‌سازند.

۱۰. روش‌های آموختنی detector/descriptor/matcher را بهینه می‌کنند، اما ارزیابی منصفانه همچنان به ،uncertainty geometry و accounting failure نیاز دارد.

۱۱. سامانه آینده غالباً هیبرید است: retrieval آموختنی، feature local قوی، verfiier هندسی و fallback قابل ممیزی.

۱۲. فصل بعد می‌تواند از ویژگی‌های منفرد به حرکت، ویدئو، ردیابی و برآورد correspondence زمانی گسترش یابد.

منابع منتخب فصل

[1] C. Harris and M. Stephens, “A combined corner and edge detector,” Alvey Vision Conference, 1988.

[2] J. Shi and C. Tomasi, “Good features to track,” IEEE Conference on Computer Vision and Pattern Recognition, 1994.

[3] E. Rosten and T. Drummond, “Machine learning for high-speed corner detection,” European Conference on Computer Vision, 2006.

[4] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005.

[5] K. Mikolajczyk et al., “A comparison of affine region detectors,” International Journal of Computer Vision, 2006.

[6] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” International Journal of Computer Vision, 2004.

[7] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-Up Robust Features (SURF),” Computer Vision and Image Understanding, 2008.

[8] J. Matas, O. Chum, M. Urban, and T. Pajdla, “Robust wide-baseline stereo from maximally stable extremal regions,” Image and Vision Computing, 2004.

[9] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, “KAZE features,” European Conference on Computer Vision, 2012.

[10] P. F. Alcantarilla, J. Nuevo, and A. Bartoli, “Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces,” British Machine Vision Conference, 2013.

[11] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, “BRIEF: Binary Robust Independent Elementary Features,” European Conference on Computer Vision, 2010.

[12] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: An efficient alternative to SIFT or SURF,” IEEE International Conference on Computer Vision, 2011.

[13] S. Leutenegger, M. Chli, and R. Siegwart, “BRISK: Binary Robust Invariant Scalable Keypoints,” IEEE International Conference on Computer Vision, 2011.

[14] A. Alahi, R. Ortiz, and P. Vandergheynst, “FREAK: Fast Retina Keypoint,” IEEE Conference on Computer Vision and Pattern Recognition, 2012.

[15] R. Arandjelovic and A. Zisserman, “Three things everyone should know to improve object retrieval,” IEEE Conference on Computer Vision and Pattern Recognition, 2012.

[16] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” IEEE Conference on Computer Vision and Pattern Recognition, 2005.

[17] S. Belongie, J. Malik, and J. Puzicha, “Shape matching and object recognition using shape contexts,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002.

[18] M. A. Fischler and R. C. Bolles, “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,” Communications of the ACM, 1981.

[19] O. Chum and J. Matas, “Matching with PROSAC—Progressive Sample Consensus,” IEEE Conference on Computer Vision and Pattern Recognition, 2005.

[20] O. Chum, J. Matas, and J. Kittler, “Locally optimized RANSAC,” DAGM Symposium, 2003.

[21] D. Barath, J. Noskova, M. Ivashechkin, and J. Matas, “MAGSAC++, a fast, reliable and accurate robust estimator,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

[22] M. Turk and A. Pentland, “Eigenfaces for recognition,” Journal of Cognitive Neuroscience, 1991.

[23] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, “Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997.

[24] C. Cortes and V. Vapnik, “Support-vector networks,” Machine Learning, 1995.

[25] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” IEEE Conference on Computer Vision and Pattern Recognition, 2001.

[26] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010.

[27] J. Sivic and A. Zisserman, “Video Google: A text retrieval approach to object matching in videos,” IEEE International Conference on Computer Vision, 2003.

[28] D. Nister and H. Stewenius, “Scalable recognition with a vocabulary tree,” IEEE Conference on Computer Vision and Pattern Recognition, 2006.

[29] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” IEEE Conference on Computer Vision and Pattern Recognition, 2006.

[30] H. Jegou, M. Douze, C. Schmid, and P. Perez, “Aggregating local descriptors into a compact image representation,” IEEE Conference on Computer Vision and Pattern Recognition, 2010.

[31] F. Perronnin, J. Sanchez, and T. Mensink, “Improving the Fisher kernel for large-scale image classification,” European Conference on Computer Vision, 2010.

[32] J. Sanchez, F. Perronnin, T. Mensink, and J. Verbeek, “Image classification with the Fisher vector: Theory and practice,” International Journal of Computer Vision, 2013.

[33] D. DeTone, T. Malisiewicz, and A. Rabinovich, “SuperPoint: Self-supervised interest point detection and description,” CVPR Workshops, 2018.

[34] M. Dusmanu et al., “D2-Net: A trainable CNN for joint detection and description of local features,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[35] J. Revaud et al., “R2D2: Repeatable and reliable detector and descriptor,” NeurIPS, 2019.

[36] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, “SuperGlue: Learning feature matching with graph neural networks,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

[37] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, “LoFTR: Detector-free local feature matching with Transformers,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021.

[38] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys, “LightGlue: Local feature matching at light speed,” IEEE/CVF International Conference on Computer Vision, 2023.

[39] Y. Wang, X. He, S. Peng, D. Tan, and X. Zhou, “Efficient LoFTR: Semi-dense local feature matching with sparse-like speed,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024.

[40] J. Edstedt et al., “RoMa: Robust dense feature matching,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024.

[41] H. Jiang et al., “OmniGlue: Generalizable feature matching with foundation model guidance,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024.

[42] G. Chen et al., “RDD: Robust feature detector and descriptor using deformable Transformer,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2025.

[43] S. W. Choo et al., “Scalable feature matching via state space modeling and sparse correlation,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2026.

[44] OpenCV Development Team, “Feature detection, description, matching, homography and stitching documentation,” accessed July 2026.

راهنمای کامپایل، فونت و اجرای آزمایش‌ها

فایل با XeLaTeX کامپایل می‌شود. فونت اصلی Amiri، فونت لاتین DejaVu Serif و فونت کد DejaVu Sans Mono است و جایگزین خودکار تعریف شده است. هیچ فایل فونتی در بسته توزیع نمی‌شود.

```
python -m venv .venv
# Windows: .venv\Scripts\activate
# Linux/macOS: source .venv/bin/activate
pip install -r requirements.txt
python generate_figures.py
python chapter09_experiments.py --output outputs
xelatex image_processing_next_decade_ch09_fa.tex
xelatex image_processing_next_decade_ch09_fa.tex
```