

**دانشگاه صنعتی امیر کبیر**  
(پلی تکنیک تهران)

گروه مستقل مهندسی رباتیک

گزارش مطالعاتی درس بینایی ربات

عنوان:

**تشخیص بیضی**

**Ellipse Detection**

استاد درس:

دکتر رضا صفابخش

پژوهشگر:

نوید خزاعی

تیر ۱۳۹۳

## چکیده

برای تشخیص بیضی، روش‌های معمول یا مبتنی بر تبدیل هاف هستند، یا بر مبنای ویژگی‌های هندسی بیضی و ابتکاری هستند. در این گزارش برخی روش‌های موجود را برشمردیم و در برخی موارد تحلیلی از چگونگی عملکرد و مشکلات و بهبودها ارائه کردیم. سه مورد از روش‌های مبتنی بر تبدیل هاف، و یک مورد از روش‌های ابتکاری را نیز به شکل جزئی‌تر بررسی کردیم و اشاره‌ای نیز به مقایسه‌های موجود در مقالات نموده‌ایم.

## فهرست مطالب

۱	مقدمه	۱
۳	الگوریتم‌های تشخیص و کارهای مرتبط	۲
۴	۱.۲ روش‌های مبتنی بر رای‌گیری یا خوشه‌بندی	۱.۲
۵	۲.۲ روش‌های ابتکاری یا ترکیبی	۲.۲
۸	۳ تبدیل هاف سریع بیضی (FEHT)	۳
۸	۱.۳ الگوریتم و استراتژی تمرکز	۱.۳
۱۰	۲.۳ تشخیص بیضی	۲.۳
۱۰	۱.۲.۳ تشخیص مراکز	۱.۲.۳
۱۲	۲.۲.۳ تشخیص جهت‌ها	۲.۲.۳
۱۲	۳.۲.۳ تشخیص نیم‌قطرها	۳.۲.۳
۱۳	۴.۲.۳ نتیجه‌گیری	۴.۲.۳
۱۵	۴ تبدیل هاف تصادفی‌شده برای بیضی (RHT)	۴
۱۵	۱.۴ خطی‌سازی	۱.۴
۱۷	۵ تشخیص بر مبنای ویژگی‌های هندسی	۵
۲۰	مراجع	۲۰

## ۱ مقدمه

تشخیص اشیا و اشکال هندسی، همواره یکی از چالش‌برانگیزترین موضوعات در بینایی ماشین بوده‌است. در این میان، دایره و بیضی از اهمیت بیشتری برخوردارند. بیضی حالت عام‌تری از دایره است چرا که دایره همان بیضی است که نیم‌قطر کوچک و نیم‌قطر بزرگ آن مساوی است. در تصاویر واقعی، افکنش‌های دایره‌های موجود ممکن است به شکل بیضی در تصویر ظاهر شوند. همچنین در بسیاری از موارد دایره‌های ایده‌آل نداریم و کشیدگی‌ها و ناهمواری‌ها سبب می‌شود تا شکل‌ها بیشتر شبیه بیضی باشند. از این رو، تشخیص بیضی‌ها در پردازش تصاویر واقعی می‌تواند بسیار موثر باشد و درک ما از تصویر را افزایش بدهد. همچنین می‌توان اشکال را ویژگی‌هایی برای توصیف شکل دانست و با توجه به کاربرد مساله، هر تصویر را با اشکال موجود توصیف کرد. با این وصف، حجم مجموعه‌ی داده‌ها کاهش داده می‌شود، بنابراین می‌توان از این مجموعه‌ی برای حل مساله‌ی مورد نظر، و یا به عنوان ورودی برای پردازش‌های آتی، استفاده نمود.

در ریاضیات، بیضی یک خم مسطح (خم محصور در صفحه‌ی اقلیدسی)، از خانواده‌ی مخروطی‌ها<sup>۱</sup> و یک مقطع مخروطی محدود است. مقطع مخروطی حاصل از قطع دادن یک صفحه با یک مخروط دایره‌ای است. خم‌های دیگری نیز از قطع کردن مخروط به دست می‌آیند که نامحدود هستند و سهمی و هذلولی نام دارند [۱]. همان‌گونه که اشاره شد، بیضی از دایره عمومی‌تر است و در ریاضیات نیز، معیاری برای سنجش انحراف بیضی از دایره‌ی معادل وجود دارد که به آن گریز از مرکز<sup>۲</sup> می‌گویند. این ارتباط، خود می‌تواند اطلاعاتی در مورد اندازه و جهت به دست دهد که در کاربردهای زیادی مورد استفاده قرار می‌گیرد. محققان کاوش‌های زیادی بر روی ویژگی‌های بیضی انجام داده‌اند که از جمله‌ی آن‌ها می‌توان به استفاده از طبیعت متقارن کانون‌ها<sup>۳</sup> برای پارامتری کردن بیضی اشاره نمود [۲]. همچنین از ارتباط توأم قطر‌ها بین وترهای موازی برای پیش‌بینی مرکز بیضی استفاده شده‌است [۳]. تشخیص دقیق بیضی در کاربردهای بی‌درنگ بسیار مهم است، چرا که دقت تشخیص می‌تواند به‌خاطر رسیدن به پاسخ‌گویی با نرخ بالاتر، کاهش یابد. از جمله کاربردهای تشخیص بیضی می‌توان به موارد زیر اشاره کرد:

- **نظارت و دیده‌بانی:** دنبال کردن چشم، لب‌خوانی، و تشخیص چهره به شدت بر مبنای تشخیص بیضی انجام می‌شود. همچنین دنبال کردن اعضای انسان می‌تواند با دنبال کردن

<sup>۱</sup> Conics

<sup>۲</sup> Eccentricity

<sup>۳</sup> Foci

چندین بیضی انجام شود که برای تشخیص برخی اعمال مانند افتادن، کاربرد دارد. استفاده از بیضی در این مورد بهتر از کادر دور شخص است چرا که بیضی جهت و شکل شخص را نیز حفظ می‌کند.

- **کشاورزی:** محاسبه‌ی اندازه و مقایسه‌ی محصولات، می‌تواند در مرتب‌سازی و چیدن خودکار مورد استفاده قرار گیرد.

- **صنعت:** گریز از مرکز سیم‌پیچ‌های فولادی می‌تواند معیاری برای کنترل کیفیت باشد.

- **زیست‌پزشکی:** سلول‌ها را می‌توان با تشخیص بیضی، تقطیع کرد و با کمک بیضی‌های چندگانه می‌توان مدل سه‌بعدی از سلول را ساخت. به عنوان نمونه‌ای دیگر، می‌توان حجم یک پروستات بیضی‌شکل را با لایه‌هایی از بیضی‌ها تخمین زد.

- **اقیانوس‌شناسی:** گرداب‌های بیضی‌شکل بسیاری در نقشه‌های دمای سطح<sup>۱</sup> اتفاق می‌افتد.

- **نقشه‌برداری فضایی:** کمان‌های بیضی‌گون در تصاویر ماهواره‌ای از کانتورهای طبیعی موجود در سیارات و نیز سازه‌های دست بشر، به چشم می‌خورند.

- **کاوش‌های فضایی:** هاله‌های نوری خورشید شکل‌های بیضی‌گون دارند و اطلاعات به‌دست آمده از اندازه و جهت آن‌ها در موقعیت‌یابی مفید است.

- **رباتیک:** در مسایل بسیاری از جمله تشخیص تابلوهای راهنمایی و رانندگی در خودروهای هوشمند، تشخیص توپ در زمین فوتبال و غیره کاربرد دارد.

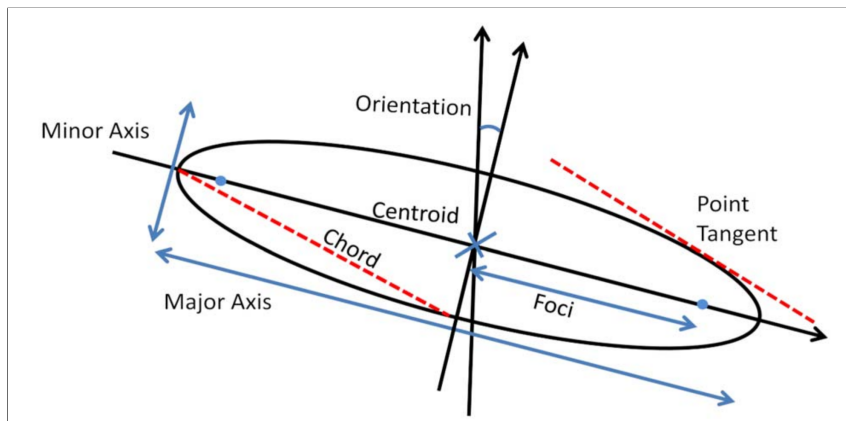
در این گزارش، در بخش دوم به بررسی روش‌های موجود برای تشخیص بیضی خواهیم پرداخت و در بخش‌های سوم تا پنجم، سه نمونه را با توجه به اهمیت کاربرد و درک الگوریتم آن‌ها به صورت جزئی‌تر بررسی خواهیم کرد.

---

<sup>۱</sup> Surface Temperature maps (STT)

## ۲ الگوریتم‌های تشخیص و کارهای مرتبط

تمامی الگوریتم‌های تشخیص بیضی، بر مبنای ویژگی‌های هندسی یکتای بیضی کار می‌کنند. برای سادگی، برخی ویژگی‌ها در شکل ۱.۲ آورده شده‌است.



شکل ۱.۲: پارامترهای بیضی (آبی) و ویژگی‌های هندسی بیضی (قرمز) [۴].

پارامترهای بیضی عبارت‌اند از مرکز (مرکز ثقل)<sup>۱</sup>، نیم‌قطر کوچک<sup>۲</sup>، نیم‌قطر بزرگ<sup>۳</sup>، جهت و کانون‌ها<sup>۴</sup>.

ویژگی‌های هندسی شامل وتر<sup>۵</sup> و تانژانت نقطه<sup>۶</sup> می‌باشد که به ترتیب حاصل از اتصال دو نقطه روی محیط بیضی و حاصل از قطع دادن مماس در یک نقطه با امتداد نیم‌قطر بزرگ می‌باشند. در ادبیات موضوع، روش‌های برازش بیضی به داده‌ها نیز از روش‌های تشخیص بیضی شمرده شده‌است. با این حال در گزارش پیش رو از بررسی این روش‌ها خودداری کردیم چرا که نسبت به روش‌های دیگر، کمتر از تکنیک‌های بینایی ماشین و پردازش تصویر استفاده می‌کنند و مساله‌ی برازش بیضی به داده‌ها، بسیار عام‌تر از تشخیص بیضی در تصاویر است.

به این ترتیب، می‌توان روش‌های باقی‌مانده را به دو دسته‌ی کلی تقسیم کرد: روش‌های مبتنی بر رای‌گیری یا خوشه‌بندی، و روش‌های ابتکاری یا ترکیبی. در ادامه به بررسی کارهای انجام‌شده در این زمینه خواهیم پرداخت و پس از آن، دو نمونه‌ی پرکاربرد از دسته‌ی اول و یک نمونه‌ی برگزیده

<sup>۱</sup> Centroid

<sup>۲</sup> Minor Axis

<sup>۳</sup> Major Axis

<sup>۴</sup> Foci

<sup>۵</sup> Chord

<sup>۶</sup> Point Tangent

از دسته‌ی دوم را معرفی خواهیم کرد.

## ۱.۲ روش‌های مبتنی بر رای‌گیری یا خوشه‌بندی

روش‌های خوشه‌بندی فازی همانند [۵]، مقاومت را فدای تشخیص سریع می‌کردند و قبل از به وجود آمدن روش مبتنی بر رای‌گیری «تبدیل هاف»<sup>۱</sup> استفاده می‌شدند. روش هاف یکی از محبوب‌ترین و پرکاربردترین الگوریتم‌ها در بینایی ماشین است که در [۶] برای تشخیص ویژگی‌های هندسی مانند خط و مخروطی‌ها معرفی شد که در درس بینایی ماشین به طور مفصل با آن آشنا شدیم. همچنین دیدیم که برای ورودی این تبدیل، به کار بردن یک عملگر لبه‌یابی مناسب، حیاتی است. در دهه‌های اخیر، تبدیل هاف بهبودهای زیادی پیدا کرده‌است تا محدودیت‌ها و کاربردهای بسیاری را از جمله تشخیص بیضی و کمان‌های بیضی‌گون پوشش دهد.

برخی از روش‌های هاف، از جمله خود تبدیل استاندارد هاف و بهبودهای بسیاری از آن، مانند هاف احتمالاتی<sup>۲</sup>، هاف تصاعدی<sup>۳</sup>، هاف ترکیبیاتی<sup>۴</sup> و هاف تقارن هندسی<sup>۵</sup>، که برای تشخیص خط بسیار موثر هستند، برای تشخیص چندین بیضی موثر نیستند. چرا که برای توصیف فضای پارامتری بیضی، نیاز به ۵ متغیر (۵ پارامتر بیضی) داریم، حال آن‌که برای فضای پارامتری خط تنها دو متغیر داشتیم. بدیهی است که سربار محاسباتی و نیاز به منابع پردازشی با افزایش پارامترها به صورت نمایی رشد می‌کند. جست‌وجو به دنبال قله‌ی موجود در چنین آرایه‌ی انباره‌ای<sup>۶</sup> خود به تنهایی زمان‌بر است. با این حال، مقاومت روش هاف در برابر پوشیدگی<sup>۷</sup> سبب شده‌است محققان به بهبود این روش برای تشخیص بیضی ادامه‌دهند.

روش‌های هاف موجود برای تشخیص چندین بیضی به ترتیب زمان پیدایش عبارت‌اند از: هاف عمومی<sup>۸</sup>[۷]، هاف خط مستقیم<sup>۹</sup>[۸]، تبدیل هاف سریع بیضی<sup>۱۰</sup>[۹] و هاف تصادفی<sup>۱۱</sup>[۱۰].

تبدیل هاف را می‌توان با استفاده از جهت‌های لبه‌ها و با ساختن جدول‌های توصیف‌گر شکل<sup>۱۲</sup>

<sup>۱</sup> Hough Transform (HT)

<sup>۲</sup> Probabilistic HT

<sup>۳</sup> Progressive PT

<sup>۴</sup> Combinatorial HT

<sup>۵</sup> Geometric Symmetry HT

<sup>۶</sup> Accumulator array

<sup>۷</sup> Occlusion

<sup>۸</sup> Generalized HT (GHT)

<sup>۹</sup> Straight Line HT (SLHT)

<sup>۱۰</sup> Fast Ellipse Hough Transform (FEHT)

<sup>۱۱</sup> Randomized HT (RHT)

<sup>۱۲</sup> R-Table

که در [۷] آورده شده است، برای هر شکلی عمومی کرد. در اصل روش هاف عمومی، تلاش می کند با شکستن شکل به قطعه های کوچک تر، فضای پارامتر را کم کند و پیچیدگی را کاهش دهد. در [۸] روشی برای تشخیص بیضی با روش هاف ارائه شده است که نیازی به آرایه های انباره ندارد. این روش مرکز بیضی ها را با توجه به اثر بیضی در SLHT و اصول نصف کردن قطر<sup>۱</sup> پیدا می کند. پارامتری در فضای تبدیل هاف معرفی شده است که به کمک آن می توان با استفاده از یک الگوریتم تشخیص خوشه های ساده، بیضی ها را پیدا کرد و نیازی به جست و جوی قله ها در آرایه های انباره نیست. پیچیدگی زمانی تابع اصلی در این روش  $O(n^2)$  است.

FEHT با روش سرکشی معمول در روش هاف تفاوت دارد، چرا که اساس آن یک روش تکراری تمرکز کردن<sup>۲</sup> است [۹]. این روش برای وقتی که بیضی هایی با اندازه های متفاوت به دفعات زیاد در تصویر ظاهر می شوند مفید است و می تواند آن ها را از هم تفکیک کند. در این گزارش به بررسی دقیق تر این روش در بخش سوم خواهیم پرداخت.

RHT بر مبنای تخمین تانژانت های محلی است تا با این کار، مساله ی غیرخطی را به یک مساله ی خطی کاهش دهد. در این روش، انبار کردن نقاط فضای پارامتری با انتخاب تصادفی چندتایی هایی از پیکسل ها و محاسبه ی پارامترهای شکلی که از این پیکسل ها عبور می کند، انجام می شود [۱۰]. در بخش چهارم از این گزارش به بررسی دقیق تر این روش نیز خواهیم پرداخت.

در [۱۱] تبدیل هاف با یک هرم تصویر ترکیب شده است و یک تشخیص بیضی مقاوم به دست آمده است. تصویر با کم ترین کیفیت، نقطه ی شروع در این روش است و با افزایش کیفیت تصویر، بیضی های کاندید توسط به روز رسانی یک هاف چندگذره<sup>۳</sup> محاسبه می شوند، به این ترتیب که نقاط حاصل از کیفیت جدید، نقاط حاصل از تصویر قبلی در فضای پارامتری را به روز رسانی می کنند. برای درک بهتر به شکل ۲.۲ مراجعه کنید. پیچیدگی زمانی تابع اصلی در این روش  $O(n^{5/2})$  است.

## ۲.۲ روش های ابتکاری یا ترکیبی

یکی از ابتکارات موجود، استفاده از فیلتر کالمن بسط داده شده<sup>۴</sup> است. از آن جا که EKF یک ابزار ریاضی برای بهبود مدل پیش گوینه بر اساس مشاهدات بهبود بخش است، به عنوان یک ابزار شهودی در بینایی ماشین استفاده می شود [۱۲]. جزییات این روش بیشتر شبیه روش های برازش

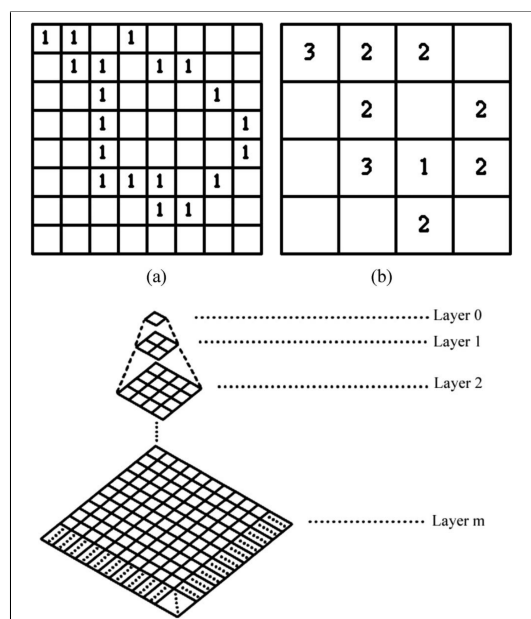
<sup>۱</sup> Diameter Bisection

<sup>۲</sup> Focusing

<sup>۳</sup> Multi pass HT

<sup>۴</sup> Extended Kalman Filter (EKF)





شکل ۲.۲: استفاده از هرم تصویر در تبدیل هاف [۱۱]

بیضی است و کمتر از ایده‌های بینایی ماشین استفاده کرده‌است. روش‌های آماری نیز برای تشخیص بیضی استفاده شده‌است. روش‌های مبتنی بر اجماع تصادفی نمونه‌ها<sup>۱</sup> [۱۳] و الگوریتم ژنتیک [۱۴] به این شکل هستند که تصویر را به زیرمجموعه‌هایی تقسیم می‌کنند. الگوریتم تصادفی، نمونه‌هایی از لبه‌ها را انتخاب می‌کند تا محتمل‌ترین مجموعه پارامتر بیضی را برای آن‌ها تشکیل دهد. سپس یک تابع هزینه برای ارزیابی وجود بیضی استفاده می‌شود که بر مبنای شمارش نقاط در نزدیکی مجموعه پارامترهای بیضی است و در نهایت یک نتیجه‌گیری برای وجود بیضی انجام می‌شود. هر دوی این الگوریتم‌ها در حضور نویز یا بیضی‌های متعدد، کارایی خود را از دست می‌دهند. برای رفع این مشکلات، شبه RANSAC [۱۵] و الگوریتم ژنتیک چند جمعیتی [۱۶] ارایه شدند.

روش دیگر، روش رشد بیضی<sup>۲</sup> [۱۷] نام دارد. این روش بر مبنای تبدیل هاف برای پیدا کردن دایره‌های مماس بر قطعات لبه‌ی موجود در تصویر است و پس از آن، طی یک روال تکراری، این قطعات با کمک اطلاعات تانژانت پیکسل‌های لبه، به بیضی رشد داده می‌شوند. اگر لبه‌ی جدایی تشخیص داده‌شود عملیات متوقف و بیضی تشخیص داده‌شده نادیده گرفته می‌شود. تفاوت این روش

<sup>۱</sup> RANdom SAMple Consensus (RANSAC)

<sup>۲</sup> Ellipse growing

با روش‌های قبلی در آن است که وقتی نیاز به دقت تقطیع بالا داشته باشیم، این روش با وجود بار محاسباتی زیاد، بهتر عمل می‌کند و در تشخیص بیضی‌های پوشیده‌شده نیز موفق‌تر بوده‌است. روش‌های ترکیبی نیز اولین بار در [۱۸] ارایه شد که از یک تبدیل هاف برای تشخیص مراکز بیضی‌های کاندید کمک گرفته‌شد و با کمک ویژگی‌های هندسی و روش کمترین مجموع مربعات، مراکز نامناسب حذف شدند.

در [۱۹]، یک روش مبتنی بر ویژگی‌های هندسی ارایه شده‌است که از قطعات لبه با اشکال خاصی استفاده می‌کند و این قطعات، لبه‌ها را به سه نوع خط، کمان و کمان کشیده<sup>۱</sup> تقسیم می‌کند. با ارایه‌ی ویژگی‌های هندسی بیضی، قاعده‌ای برای تشکیل بیضی از این قطعات ارایه شده‌است که در این گزارش به بررسی بیشتر این روش در بخش پنجم خواهیم پرداخت.

---

<sup>۱</sup> Extended arc

### ۳ تبدیل هاف سریع بیضی (FEHT)

در این بخش به بررسی جزئی تر روش ارایه شده در [۹] می پردازیم. همان گونه که در بخش اول گفته شد، تبدیل هاف به روش معمول برای تشخیص بیضی نیاز به فضای پارامتری پنج بعدی دارد. بنابراین، این روش از نظر محاسباتی بسیار سنگین می شود و از نظر مصرف حافظه نیز کارا نیست.

یکی از روش های کارا برای محاسبه ی پارامترها، شکستن پروسه ی تشخیص به مراحل کوچک تر و استفاده از اطلاعات گرادیان است که کمک می کند در هنگام شناسایی شی، یکی از پارامترها را تشخیص دهیم [۲۰].

از سوی دیگر، الگوریتم های تمرکزی وجود دارند که اجازه می دهند بیشینه در فضای پارامتر را با محاسبات و مصرف حافظه ی کمتری نسبت به روش جست و جوی سرکشی<sup>۱</sup> معمول در فضای کامل پارامتر، پیدا کنیم [۲۱]. در این میان FHT به دلیل سادگی محاسبات و قاعده مند بودن معمول تر از روش های دیگر است. هرچند اشکالاتی نیز دارد که ناشی از استراتژی دنبال شده در آن است و زمانی که تصویر شامل چند شکل با اندازه های متفاوت باشد، مشکل زا می شود.

در روش ارایه شده در [۹]، الگوریتم تمرکز جدیدی پیشنهاد شده است که مشکلات FHT را رفع کرده است. با تجزیه ی فضای پارامتر و ترکیب آن با این الگوریتم، الگوریتم FCHT برای دایره و FEHT برای بیضی ارایه شده است. این روش برای پیدا کردن مراکز شکل ها، حذف جواب های نامناسب به محض پیدا شدن برای جلوگیری از دخالت در محاسبات و برای برچسب گذاری نقاط شکل های مختلف جهت این که در مراحل بعدی تنها نقاط اثرگذار در پیدایش همان شکل استفاده شوند، محاسبات کمتری دارد.

#### ۱.۳ الگوریتم و استراتژی تمرکز

الگوریتم های تمرکز، به دنبال آن نواحی از فضای پارامتر می گردند که مقادیر بیشینه را در خود ذخیره کرده اند تا بار محاسبات را کمتر کنند. دسته ای از این الگوریتم ها یک روال تخمین سلسله مراتبی را دنبال می کنند و در هر گام، بخشی از فضای پارامتر را محاسبه می کنند و تصمیم می گیرند که به ادامه ی تحقیق کردن در این زیرناحیه بپردازند یا خیر. در FHT نیز جواب طی یک روال سلسله مراتبی تعیین می شود که در محل تلاقی چند ابر صفحه قرار دارد. در روش ارایه شده،

---

<sup>۱</sup> Polling-search

جواب در یک فضای پارامتر دو بعدی خواهد بود که به شرح جزییات آن خواهیم پرداخت. همچنین در این روش، جواب در محل تلاقی دسته‌ای از خطوط خواهد بود. تمرکز به سمت جواب، طی یک پروسه‌ی تقسیم و درج در فضای پارامتر حاصل می‌شود. برای این کار یک مربع در فضای پارامتر ایجاد می‌شود و به چهار قسمت تقسیم می‌شود. هر یک‌چهارم به تعدادی خط قابل ارزیابی (که از مقدار آستانه‌ای بیشتر باشد) که آن‌جا را پیمایش می‌کنند تقسیم می‌شود. تکرار بازگشتی این عمل، جواب را پیدا می‌کند.

برای یک خط، تصمیم‌گیری این که آیا از مربع خاصی عبور می‌کند را این‌گونه انجام می‌دهیم که فاصله‌اش با مرکز مربع را حساب می‌کنیم. اگر این فاصله از شعاع دایره‌ی محیطی مربع کم‌تر بود، آن خط این مربع را پیمایش می‌کند. برای این کار تنها نیاز است مرکز مربع، طول ضلع آن و ضرایب توصیف‌کننده‌ی خط را بدانیم. در ادامه خواهیم دید این ضرایب، مختصات نقاط در تصویر و بردارهای گرادیان متناظرشان خواهد بود. عبارتی که فاصله‌ی یک خط به معادله‌ی  $A_2y + A_1x + A_0 = 0$  را از مربع  $c = (c_x, c_y)$  می‌دهد به این شکل است:

$$d_c = A_2c_y + A_1c_x + A_0 \quad (۱.۳)$$

که در آن اندازه‌های ضرایب به‌گونه‌ای نرمال شده‌است که  $(A_2)^2 + (A_1)^2 = 1$  باشد. معادله‌ی FHT به‌گونه‌ای است که پس از محاسبه‌ی فاصله تا مرکز یک مربع، فاصله تا یک‌چهارم‌های آن مربع با عملیات کاهش‌یافته‌ی جمع و شیف‌ت قابل محاسبه است. از سوی دیگر برای کاهش محاسبات، یک بردار حضور<sup>۱</sup> تشکیل می‌شود که نشان می‌دهد هر خط در فضای پارامتر، کدام یک‌چهارم را پیمایش می‌کند (یک بیت از بردار به ازای هر خط).

استراتژی تمرکز مشخص می‌کند که چگونه یک‌چهارم‌ها را تقسیم کنیم و درخت ایجادشده را چگونه بررسی کنیم. استراتژی اول سطح، حافظه‌ی زیادی مصرف می‌کند و از طرفی، چون قاعده‌ی خاصی برای مشخص کردن این که کدام نود بهتر است باز شود نداریم، حرکت به سوی جواب ممکن است کند باشد. استراتژی اول عمق، کمی بهبود حاصل می‌شود اما همچنان نداشتن قاعده برای انتخاب بهترین نود از نودهای کاندید برای بسط دادن، کارایی را کم می‌کند. یکی از بهبودهای پیشنهادی این است که نودی که تعداد خط بیشتری را شامل شده‌است انتخاب شود که این کار در مواردی که چندین جواب در فضای پارامتر موجود است، ممکن است فشار انتخاب یک جواب را بالا ببرد.

---

<sup>۱</sup> Presence vector

برای حل مشکلات مطرح در استراتژی، یک الگوریتم تمرکز جدید معرفی شده است که بهترین عمق و سطح را انتخاب می کند. با شروع از سطح اول، همه ی نودهای ممکن تولید می شوند، ولی تنها نودهایی بررسی می شوند که از حاصل ضرب مقدار بیشینه با یک ضریب معین که ضریب تمیز<sup>۱</sup> نامیده می شود، بیشتر باشند.

### ۲.۳ تشخیص بیضی

نقاط روی یک بیضی، در رابطه ی زیر صدق می کنند:

$$\frac{[(x_p - x_0) \cos \phi + (y_p - y_0) \sin \phi]^2}{a^2} + \frac{[(x_p - x_0) \sin \phi - (y_p - y_0) \cos \phi]^2}{b^2} = 1 \quad (2.3)$$

که در آن  $x_0$  و  $y_0$  مختصات مرکز،  $\phi$  زاویه ی جهت و  $a$  و  $b$  نیم قطرهای هستند. به جای استفاده از روش معمول در FHT که نیاز به یک فضای پنج بعدی دارد و حافظه ی زیادی مصرف می کند و پردازش سنگینی دارد، از اطلاعات بردار مشتق استفاده شده است تا محاسبه ی پارامترها به چند مرحله شکسته شود. مراحل ارایه شده به این شکل است:

۱. تشخیص مرکز با استفاده از الگوریتم تمرکزی که نقطه ای را پیدا می کند که در آن دسته ای

از خطوط هم را قطع می کنند.

۲. تولید زاویه ی  $\phi$  و نرخ  $a^2/b^2$  با توجه به اطلاعات مرحله ی قبل و بردار گرادیان شکلها.

۳. محاسبه ی نیم قطرهای  $a$  و  $b$  از معادله ی بیضی.

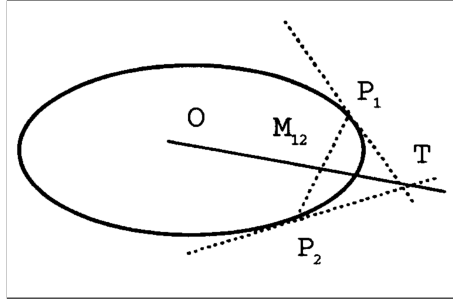
### ۱.۲.۳ تشخیص مراکز

با فرض این که  $P_1 = (x_1, y_1)$  و  $P_2 = (x_2, y_2)$  دو نقطه از بیضی با شیبهای<sup>۲</sup>  $\xi_1$  و  $\xi_2$  باشند، که در نقطه ی  $T_{12}$  که در شکل ۱.۳ نشان داده شده است، یکدیگر را قطع می کنند. خطی که نقطه ی  $T_{12}$  و نقطه ی وسط  $M = [(x_1 + x_2)/2, (y_1 + y_2)/2]$  می گذرد، از مرکز بیضی عبور می کند.

بنابراین با استفاده از جفت نقاطی که شیبهای موازی ندارند می توان دسته ای از خطوط تولید کرد که هم را در مرکز بیضی قطع می کنند. با اعمال الگوریتم FHT بر روی این فضا، مرکز بیضی به دست می آید. در این مرحله هر خط با ضرایب نرمال شده ی متناظر با معادله ی ۱.۳ تعریف می شود:

<sup>۱</sup> Discrimination factor

<sup>۲</sup> Slope



شکل ۱.۳: تولید دسته‌ای از خط‌ها برای بیضی [۹]

$$A_0^{p_1 p_2} = \frac{t_1 m_2 - t_2 m_1}{\Gamma_{12}}, \quad A_1^{p_1 p_2} = \frac{t_2 - m_2}{\Gamma_{12}}, \quad A_2^{p_1 p_2} = \frac{t_1 - m_1}{\Gamma_{12}} \quad (۳.۳)$$

که در آن:

$$t_1 = \frac{y_1 - y_2 - x_1 \xi_1 + x_2 \xi_2}{\xi_2 - \xi_1}, \quad t_2 = \frac{\xi_2 \xi_1 (x_2 - x_1) - y_2 \xi_1 + y_1 \xi_2}{\xi_2 - \xi_1},$$

$$m_1 = \frac{x_1 + x_2}{2}, \quad m_2 = \frac{y_1 + y_2}{2},$$

$$\Gamma_{12} = \sqrt{(t_2 - m_2)^2 + (t_1 - m_1)^2} \quad (۴.۳)$$

شیب نقاط از اطلاعات گرادیان در آن‌ها به دست می‌آید. به این صورت، اگر  $\theta$  زاویه‌ی گرادیان در نقطه‌ی  $P$  باشد، زاویه‌ی گرادیان در آن نقطه برابر است با:

$$\xi_p = \theta_p - \frac{\pi}{2} \quad (۵.۳)$$

به این ترتیب می‌توان با پیچیدگی زمانی برابر  $O(n^2)$  مرکز بیضی‌ها را پیدا کرد. جزییات پیچیدگی زمانی برای هر بخش و نیز شروط انتخاب نقاط در شرایط خاص در [۹] آمده‌است که بیان آن از حوصله‌ی این بخش خارج است.

### ۲.۲.۳ تشخیص جهت‌ها

با محاسبه‌ی مشتقات جزئی معادله‌ی بیضی بدون دوران، هر نقطه‌ی  $P$  روی کانتور بیضی در معادله‌ی زیر صدق می‌کند:

$$\frac{a^2}{b^2} = \frac{(x_p - x_0)}{(y_p - y_0)} \tan \theta_p \quad (۶.۳)$$

اگر بیضی به اندازه‌ی  $\phi$  دوران داده شده‌باشد، معادله‌ی ۶.۳ به این صورت درمی‌آید:

$$\frac{a^2}{b^2} = \frac{(x_p - x_0) \cos \phi + (y_p - y_0) \sin \phi}{(x_p - x_0) \sin \phi - (y_p - y_0) \cos \phi} \tan(\theta_p - \phi) \quad (۷.۳)$$

با توجه به آن که مختصات مراکز را از مرحله‌ی قبل به‌دست آورده‌ایم، در این بخش می‌توانیم از یک فضای دو بعدی استفاده کنیم که در آن هر نقطه از بیضی برای جهتی خاص رای می‌دهد. پس از آن که مقادیر بیشینه را پیدا کردیم، جهات مختلف بیضی‌ها در تصویر را به‌دست آورده‌ایم و با توجه به معادله‌ی ۷.۳، نسبت نیم‌قطرها نیز به‌دست می‌آید.

باید به این نکته دقت داشت که برای صرفه‌جویی در استفاده از توان پردازشی، در این فاز تنها از نقاطی استفاده می‌شود که در پیدا کردن مراکز بیضی‌ها نقش داشته‌اند. به همین ترتیب اگر بیضی‌های هم‌مرکز ولی با جهات متفاوت وجود داشته باشد، در این فاز رای‌گیری نقاطی که در پیدا کردن یک مرکز مشترک مشارکت داشته‌اند، هم‌اکنون به جهات و نرخ‌های نیم‌قطرهای متفاوت رای می‌دهند.

### ۳.۲.۳ تشخیص نیم‌قطرها

برای همه‌ی نقاطی که در مرحله‌ی قبل مشارکت کرده‌اند، دورانی به اندازه‌ی  $\phi$  انجام داده می‌شود و به‌جای نرخ نیم‌قطرها،  $h$  قرار داده می‌شود. پس مقادیر نیم‌قطرها به این گونه به‌دست می‌آید:

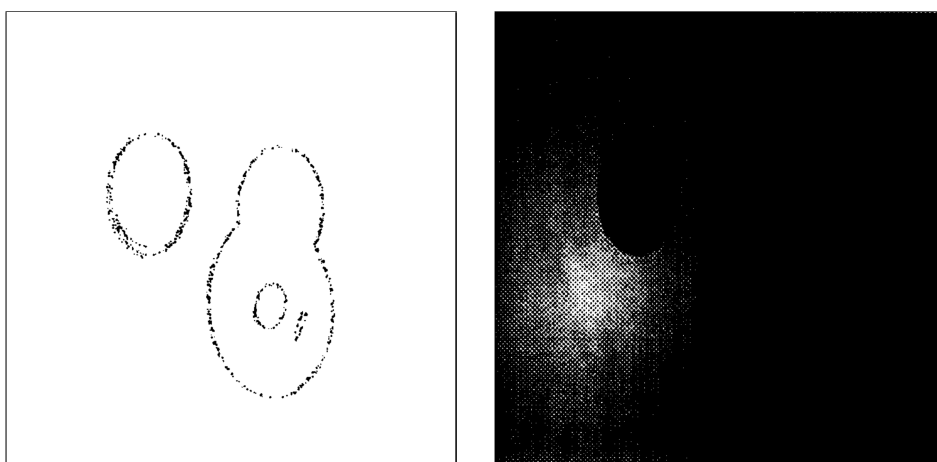
$$b = \sqrt{h^{-1}(x_p - x_0)^2 - (y_p - y_0)^2}$$

$$a = \sqrt{(x_p - x_0)^2 + h(y_p - y_0)^2} \quad (۸.۳)$$

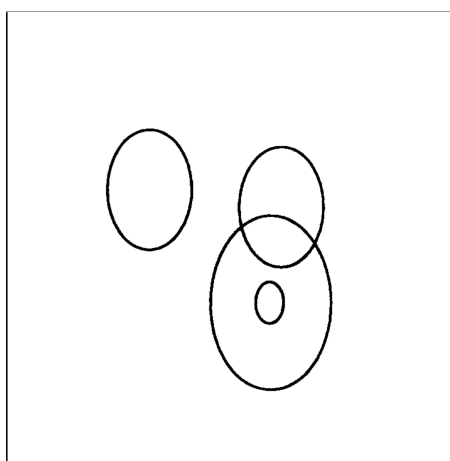
در این جا نیز برای کاهش پیچیدگی می توان از این نکته سود برد که نقاطی که در فاز پیش رای داده اند به نسبت خاصی از مقادیر نیم قطر ها رای داده اند و در پیدا کردن بیشینه ها این مطلب برای پیدا کردن زوج نیم قطر های نظیر، مفید است.

### ۴.۲.۳ نتیجه گیری

برای یک تصویر  $512 \times 512$  نزدیک به 50% از نقاط در روند اجرا حذف شده اند. برای لبه یابی از عملگر سوبل استفاده شده است و ضریب تمیز برابر با 0.8 قرار داده شده است.



شکل ۲.۳: راست: تصویر اصلی. چپ: تصویر لبه یابی شده [۹]



شکل ۳.۳: بیضی های تشخیص داده شده [۹]



نودهای فعال	زمان (s)	افزایش سرعت (%)
4	100	0
8	71	29
12	79	21
16	75	25

جدول ۱.۳: مقایسه‌ی کارایی با هاف بیضی

طبق آزمایشات [۹]، این روش در مقایسه با هاف بیضی در جدول ۱.۳ مقایسه شده‌است. تعداد نودها، حداکثر تعداد نودهای قابل بررسی در استراتژی تمرکز است.

## ۴ تبدیل هاف تصادفی شده برای بیضی (RHT)

یک سال پس از ایده‌ی [۹]، بهبود دیگری برای استفاده از هاف در تشخیص بیضی انجام گرفت که بر پایه‌ی هاف تصادفی شده [۲۲] طراحی شد [۱۰]. برای استفاده از RHT باید شکل با یک معادله‌ی خطی توصیف شود، در ادامه به بررسی RHT می‌پردازیم و سپس ایده‌ی ارائه شده برای توصیف بیضی با یک معادله‌ی خطی را بررسی می‌کنیم.

روش RHT، نقاط را در فضای پارامتر با انتخاب کردن لیست‌های  $n$  تایی از پیکسل‌های تصویر و محاسبه‌ی پارامترهای شکل عبورکننده از آن‌ها، روی هم جمع می‌کند.

بیضی با معادله‌ی زیر تعریف می‌شود:

$$a(x-p)^2 + 2b(x-p)(y-p) + c(y-q)^2 = 1 \quad (۱.۴)$$

با این محدودیت که:

$$ac - b^2 > 0 \quad (۲.۴)$$

این معادله غیرخطی است [۱۰].

### ۱.۴ خطی‌سازی

برای خطی‌سازی روش زیر ارائه شده است:

با انتخاب سه پیکسل از تصویر، تانژانت در آن نقاط محاسبه می‌شود. این کار با تعریف یک همسایگی کوچک در اطراف نقطه و پیدا کردن بهترین خط منطبق به آن پیکسل‌ها (با روش کمترین مربعات) انجام می‌شود. سه پیکسل  $x_1, x_2, x_3$  و تانژانت محاسبه شده، ورودی‌های الگوریتم هستند. محاسبه‌ی پارامترهای بیضی به دو مرحله تقسیم می‌شود: پیدا کردن مراکز، و پیدا کردن سه پارامتر دیگر.

برای پیدا کردن مراکز از روش [۲۳] استفاده شده است که بر اساس ویژگی‌های هندسی بیضی است. به این ترتیب که دو نقطه روی بیضی را در نظر گرفته و نقطه‌ی میانی آن‌ها  $m$  نامیده می‌شود. همچنین محل تلاقی تانژانت‌های آن‌ها  $t$  نامیده می‌شود. مرکز بیضی بر روی خط  $\overline{tm}$  قرار خواهد داشت. این روش را در شکل ۱.۳ نیز مشاهده نمودیم. به همان روشی که در بخش ۱.۲.۳ نشان داده شد استفاده شده است.

سه پارامتر باقی مانده به این روش تخمین زده می شوند که معادله ی ۱.۴ به این صورت کاهش می یابد:

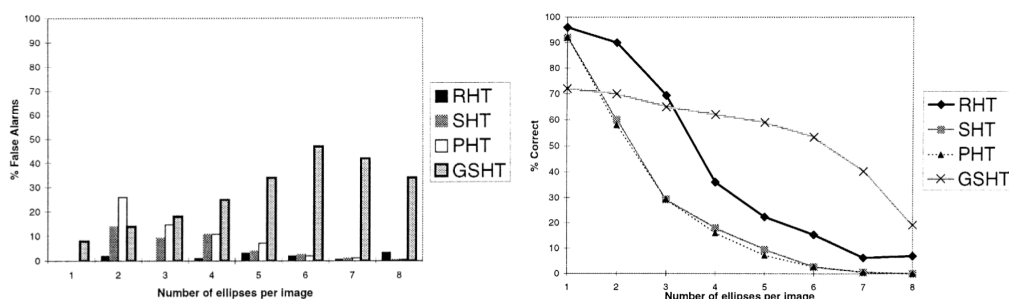
$$ax^2 + 2bxy + cy^2 = 1 \quad (۳.۴)$$

به این ترتیب با در نظر گرفتن مختصات سه نقطه، سه معادله ی خطی داریم:

$$\begin{bmatrix} x_1^2 & 2x_1y_1 & y_1^2 \\ x_2^2 & 2x_2y_2 & y_2^2 \\ x_3^2 & 2x_3y_3 & y_3^2 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad (۴.۴)$$

حل این دستگاه معادلات، سه پارامتر باقی مانده به دست می آیند. اگر بیضی معتبر باشد، پارامترهای به دست آمده در محدودیت ۲.۴ صدق می کنند. در غیر این صورت، یا سه نقطه بر روی بیضی قرار ندارند و یا تخمین تانژانت دقیق نبوده است. در این صورت سه پیکسل جدید انتخاب می شود و الگوریتم ادامه پیدا می کند.

این روش در برابر نویز و تعداد زیاد بیضی در تصویر مقاوم است. نتایج الگوریتم با سه روش دیگر، SHT [۲۴]، PHT [۲۳]، GSHT [۲۵] در شکل ۱.۴ آورده شده است.



(ب) درصد تشخیص غلط

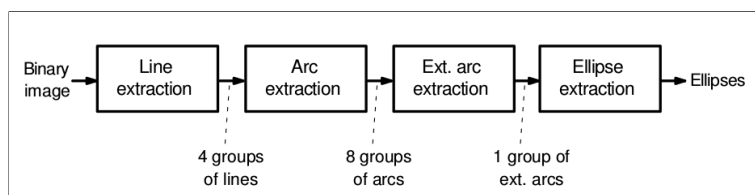
(آ) درصد دقت با چند بیضی

شکل ۱.۴: مقایسه ی دقت و تشخیص غلط [۱۰]

در [۲۶] بر مبنای همین روش، بیضی ها تشخیص داده می شوند و با خوشه بندی، برای هر دسته بیضی با پارامترهای شبیه، یک بیضی نماینده انتخاب می شود که در تشخیص بیضی در تصاویر طبیعی کاربرد دارد.

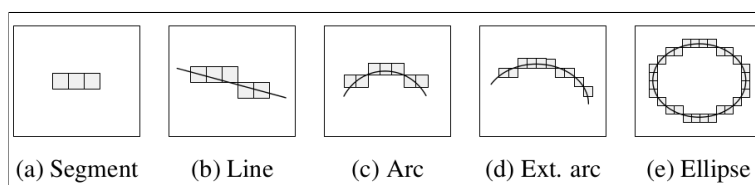
## ۵ تشخیص بر مبنای ویژگی‌های هندسی

در [۱۹] روشی برای تشخیص بیضی بر مبنای ویژگی‌های هندسی و بدون استفاده از تبدیل هاف ارایه شده‌است که نسبت به روش‌های مبتنی بر هاف، حافظه‌ی کمتری مصرف می‌کند. این روش تصویر لبه‌ها را دریافت می‌کند و در چهار مرحله بیضی‌ها را تشخیص می‌دهد. در مرحله‌ی اول قطعات کوچک خط را استخراج می‌کند و در مرحله‌ی دوم با ترکیب این خطوط، کمان‌های کوچک را تشخیص می‌دهد و سپس با ترکیب این قطعات در مرحله‌ی سوم، خم‌های کشیده را تشکیل می‌دهند که برای تشخیص بیضی مورد استفاده قرار می‌گیرند.

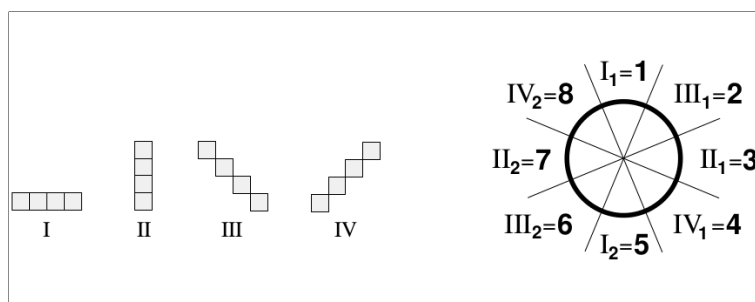


شکل ۱.۵: مراحل تشخیص در [۱۹]

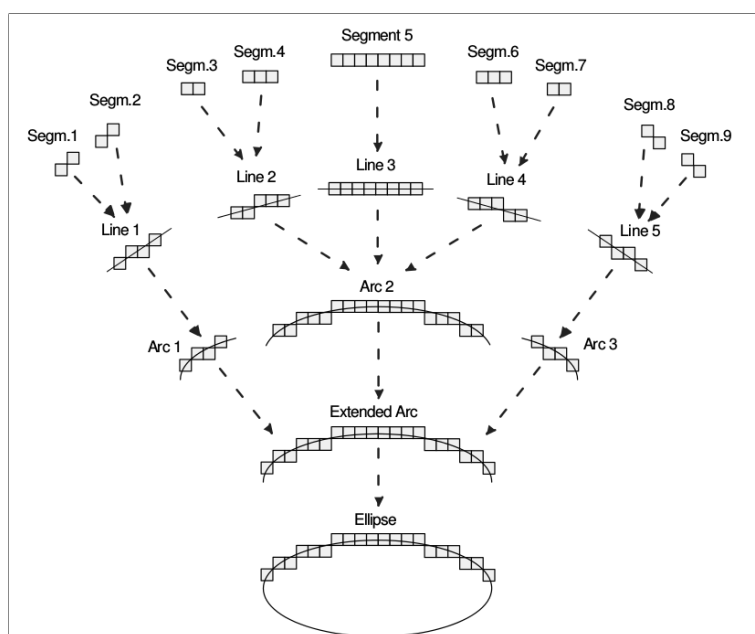
برای تشخیص، از شکل‌ها معرفی شده در شکل ۲.۵ استفاده شده‌است. از ترکیب گروه‌های خط و کمان که در شکل ۳.۵ آورده شده‌اند، اشکال یادشده ساخته می‌شوند و با شرط تجاوز نکردن از زاویه‌ی آستانه‌ای، به هم می‌پیوندند تا بیضی تشکیل شود (شکل ۵.۵).



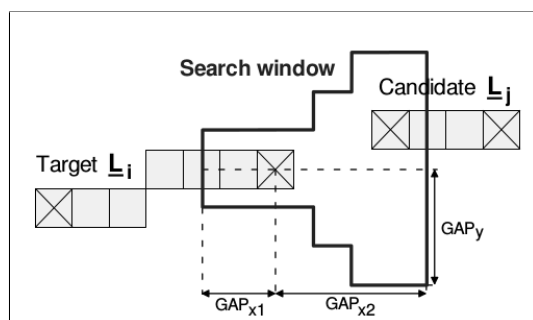
شکل ۲.۵: اشکال استخراج شونده در حین تشخیص [۱۹]



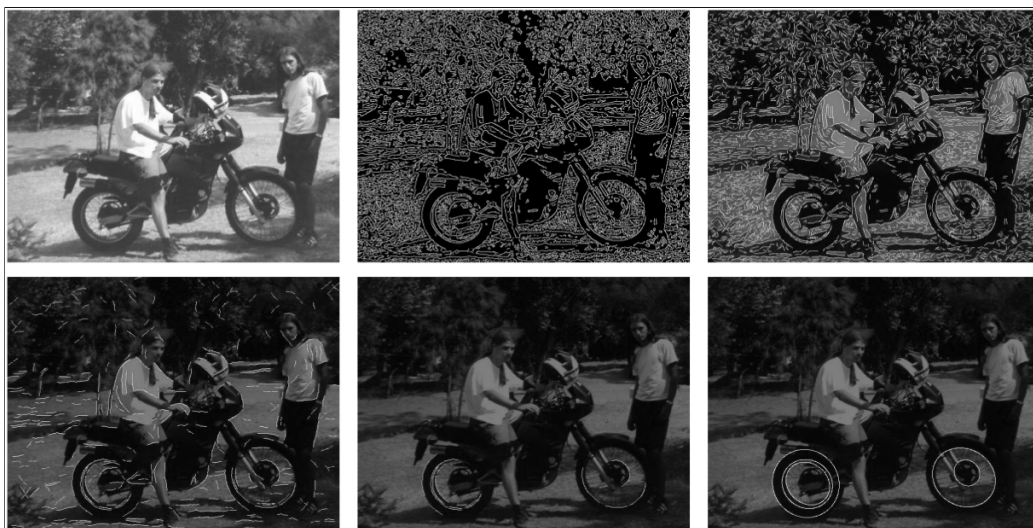
شکل ۳.۵: گروه‌های خط و کمان [۱۹]



شکل ۴.۵: گروه‌های خط و کمان [۱۹]



شکل ۵.۵: پنجره‌ی لغزان برای یافتن کمان [۱۹]



شکل ۶.۵: نتیجه‌ی نهایی [۱۹]

- [1] <http://en.wikipedia.org/wiki/Ellipse>
- [2] G. Song and H. Wang, "A fast and robust ellipse detection algorithm based on pseudo-random sample consensus," in Proceedings of the 12th international conference on Computer analysis of images and patterns. Springer-Verlag, 2007, pp. 669–676.
- [3] D. Barwick, "Very fast best-fit circular and elliptical boundaries by chord data," IEEE transactions on pattern analysis and machine intelligence, pp. 1147–1152, 2008.
- [4] Wong, C. Y., S. C. F. Lin, T. R. Ren, and N. M. Kwok. "A survey on ellipse detection methods," In Industrial Electronics (ISIE), 2012 IEEE International Symposium on, pp. 1105-1110. IEEE, 2012.
- [5] R. Dave, "Generalized fuzzy c-shells clustering and detection of circular and elliptical boundaries," Pattern Recognition, vol. 25, no. 7, pp. 713–721, 1992.
- [6] R. Duda and P. Hart, "Use of the hough transformation to detect lines and curves in pictures," Communications of the ACM, vol. 15, no. 1, pp. 11–15, 1972.
- [7] D. Ballard, "Generalizing the hough transform to detect arbitrary shapes," Pattern recognition, vol. 13, no. 2, pp. 111–122, 1981.
- [8] P. Nair, A. Saunders et al., "Hough transform based ellipse detection algorithm," Pattern Recognition Letters, vol. 17, no. 7, pp. 777–784, 1996.
- [9] N. Guil and E. Zapata, "Lower order circle and ellipse hough transform," Pattern Recognition, vol. 30, no. 10, pp. 1729–1744, 1997.
- [10] R. McLaughlin, "Randomized hough transform: Improved ellipse detection with comparison1," Pattern Recognition Letters, vol. 19, no. 3-4, pp. 299–305, 1998.
- [11] C. Chien, Y. Cheng, and T. Lin, "Robust ellipse detection based on hierarchical image pyramid and hough transform," JOSA A, vol. 28, no. 4, pp. 581–589, 2011.
- [12] J. Porrill, "Fitting ellipses and predicting confidence envelopes using a bias corrected kalman filter," Image and Vision Computing, vol. 8, no. 1, pp. 37–41, 1990.
- [13] R. Bolles and M. Fischler, "A ransac-based approach to model fitting and its application to finding cylinders in range data," in International Joint Conference on Artificial Intelligence, pp. 637–643, 1981.

- [14] P. Yin, "A new circle/ellipse detector using genetic algorithms," *Pattern Recognition Letters*, vol. 20, no. 7, pp. 731-740, 1999.
- [15] G. Song and H. Wang, "A fast and robust ellipse detection algorithm based on pseudo-random sample consensus," in *Proceedings of the 12th international conference on Computer analysis of images and patterns*. Springer-Verlag, pp. 669-676, 2007.
- [16] N. Kharma, H. Moghnieh, J. Yao, Y. Guo, A. AbuBaker, J. Laganier, G. Rouleau, and M. Cheriet, "Automatic segmentation of cells from microscopic imagery using ellipse detection," *Image Processing, IET*, vol. 1, no. 1, pp. 39-47, 2007.
- [17] K. Kanatani and N. Ohta, "Automatic detection of circular objects by ellipse growing," *International Journal of Image and Graphics*, vol. 4, no. 1, pp. 35-50, 2004.
- [18] S. Tsuji and F. Matsumoto, "Detection of ellipses by a modified hough transformation," *Computers, IEEE Transactions on*, vol. 100, no. 8, pp. 777-781, 1978.
- [19] L. Libuda, I. Grothues, and K. F. Kraiss, "Ellipse detection in digital image data using geometric features," in *Advances in Computer Graphics and Computer Vision*. Springer Berlin Heidelberg, pp. 229-239, 2007.
- [20] C. Ho and L. Chen, "A fast ellipse/circle detector using geometric symmetry," *Pattern Recognition*, vol. 28, no. 1, pp. 117-124, 1995.
- [21] H. Li, M. A. Lavin and R. J. Le Master, "Fast Hough transform: A hierarchical approach," *CVGIP* 36, pp. 139-161, 1986.
- [22] L. Xu, E. Oja, and P. Kultanen, "A new curve detection method: randomized Hough transform (RHT)," *Pattern recognition letters*, vol. 11, no. 5, pp. 331-338, 1990.
- [23] H. K. Yuen, J. Illingworth, and J. Kittler, "Detecting partially occluded ellipses using the Hough transform," *Image and Vision Computing*, vol.7, no. 1, pp. 31-37, 1989.
- [24] N. Kiryati, Y. Eldar, and A. M. Bruckstein, "A probabilistic Hough transform," *Pattern recognition*, vol. 24, no. 4, pp. 303-316, 1991.
- [25] C. T. Ho, and L. H. Chen, "A fast ellipse/circle detector using geometric symmetry," *Pattern Recognition*, vol. 28, no. 1, pp. 117-124, 1995.
- [26] C. A. Basca, M. Talos, and R. Brad, "Randomized Hough transform for ellipse detection with result clustering," in *Computer as a Tool, The International Conference on, EUROCON*, vol. 2, pp. 1397-1400, 2005.