

(۱) ابتدا به تعریف توانسته بودن توجه می کنیم. این توانایی است که در آن مقادیر

مقدور را به مجموعه ای محدود از مقادیر ممکن تقریب می زنیم. \Rightarrow در مورد توانایی در اینجا

یعنی تقریب دادن مقادیر بدلیل آنها به یک مجموعه ای محدود از مقادیرها.

$$\text{محدوده کا مقادیر} = \frac{\text{تعداد} \times \text{سطوح توانسته} - \text{توانسته مقدار سبک} - \text{بسیار مقدار سبک و شنا}}{\text{روشنای}}$$

$$= \frac{(255 - 0)}{64} \approx 3.984375$$

نکته: هر چه مقدار سطوح توانسته بیشتر باشد، یعنی تصویر بالاتر است و هر چه مقدار سطوح

توانسته کمتر باشد، موجب از دست رفتن اطلاعات و کاهش کیفیت تصویر می شود.

به طور کلی در دوربین های امروزی مقادیر ۸ بیت هستند (یعنی به ۲۵۵ سطح توانسته سه اند).

۲- می دانیم به هر چه سرعت shutter بیشتر باشد یعنی لحظات کمتری طول می کشد تا تصویر

ثبت شود، پس وقت و جزئیات تصویر بهتر هستند.

تعریف shutter speed: میزان زمانی که حسگر دوربین بازمی ماند تا نور به آن برسد.

برای پرنده‌ی در حال حرکت اگر سرعت shutter زیاد باشد، می تواند حرکت پرنده را بصورت کاملاً

متوقف نشان دهد و در نتیجه تصویر بسیار واضحی خواهد بود. اگر سرعت shutter کم باشد،

حرکت پرنده در تصویر به صورت محاسبه‌ی کمی حرکتی ظاهر می شود و تصویر ممکن است مات و نا واضح باشد.

سرعت shutter مناسب برای پرنده‌ی در حال پرواز، بستگی به عواملی مثل سرعت پرنده و

فاصله اش از دوربین و ابزاری که برای نورکشی دارد، اما بطور کلی سرعتی در حدود ۱/۱۰۰۰ الی ۱/۱۰۰۰۰

مانند می تواند مناسب به نظر برسد.

پرنده ها با توجه به نوع آن ها سرعت های متفاوتی دارند (ملاحظه کن) 50 km/h
 (لنگر کاوی) 100 km/h
 (عقاب) 320 km/h | اگر بطور میانگین پرندگی با سرعت 100 km/h را

در نظر بگیریم \rightarrow در هر ۱۰۰۰۰ ثانیه، ۲۷۷ cm حرکت خواهند کرد.

یعنی نسبتاً عکس واضحی خواهد بود.

Shutter speed:

- ۳ -

سرعت shutter باید اتراس باید تا براند و فرج بیشتر را با جزئیات بهتر ثبت کند.

DOF:

با توجه به اینکه در حالت فکلی هم یوز بلیک و هم پس زمینه هر دو آره هستند پس بیشتر می رسد احتمال محدودی مکن الان بسیار کوچیک است، چیزی که محدود یک نقطه. با توجه به شرایط فکلی باید محدودی مکن نسبت به چیزی که هست اتراس باید، اما باید وقت شود به این محدودی در انتها باید سطحی در محدود یک یوز بلیک باشد و این محدودی مکن نسبتاً کوچکی است.

نسبتی که هم به احتمالاً در عکس در حالت فکلی بسیار روبر است و باید کوچکی شود (اما مقداری که پیرامونند پس از توجه شدن همچنان اندازه‌های در عکس را در دسته‌های large قدری دهد چرا که ما داریم آن را تنها به اندازه‌ای کوچکی نمی‌م محدودی مکن از نقطه به یوز بلیک برسد).

پس داریم نسبت به حالت فکلی مکن دید را اتراس می‌دهیم و در عکس را کوچکی نمی‌م (سایر چیزی که محدود $f/2.8$ در انتها خوب باشد).

Date:

Sub:

اینطرا با لایف می شود ، تعداد نور عبوری از لنز به حسگر ها می رسد کاهش یابد .

در مرحله دوم از لنز استفاده می کنیم ، تا نور کمی عبور کند از دریچه و افتد در لنز و لنز تصویر نهایی

را افزایش دهیم .

در نهایت یونز لنز را واضح و پس زمینه را تاریک کرده ایم .

FOV:

وسعت منظره ای که ثبت شده نیز منظره ای که در دو لنز است ، به منظره افزایش می یابد ، FOV

(field of view) (میدان دید) را افزایش دهیم . برای اینطرا باید از لنز بزرگتری

استفاده کنیم به فاصله های فازی که نسبت به حالت فعلی کمتر باشند . هر چه برای افزایش

AFOV با توجه به فرمول زیر ، f (فاصله فازی) کمتر نیاز است .

$$AFOV = 2 \tan^{-1} \left(\frac{h}{2f} \right)$$

۵ - الف) این تفاوت های واضح این است که تصویر را با opencv

load ایم و بخوانیم با opencv و matplotlib نمایش دهیم، رنگ بندی که این تفاوت

است، چرا؟ چون opencv تصاویر را BGR (Blue, Green, Red) می خواند

اما matplotlib با تصاویر بصورت RGB کار می کند، پس ترتیب کانال های این تفاوت

است، حال اگر بخوانیم با opencv بخوانیم اما در matplotlib درست بسیم چطور کنیم؟

می توانیم تصویر را به حالت RGB، convert کنیم، بازستور ←

$I_rgb = cv2.cvtColor(I, cv2.COLOR_BGR2RGB)$

۲) این تغییر از تفاوت های matplotlib در نمایش تصاویر، در نمودار بودن آن است به طور

اندازه های را در اطراف آن نمایش می دهد.

تفاوت ۳) تفاوت دیگر این ها این است که در opencv به طور پیش فرض اجازه می دهیم اندازه

و پان کردن (حالت تصویر به مقدار بقیه موقعیت آن درون پنجه) می دهد، اما در matplotlib کاربر

به راحتی امکان ایجاد بقیه است را ندارد (در اندازه و جهت دادن)

Date:

Sub:

نحوه نمایش و پیاده سازی

④ OpenCV برای نمایش تصویر از یک پنجره استفاده می کند ، اگر از دستور

cv2.waitKey استفاده شود ، تصویر قابل دیدن نیست و در انتها باید از دستور

cv2.destroyAllWindows() استفاده کرد .

اما نحوه نمایش در matplotlib به این شکل نیست ، بعد از کشیدن در یک figure

استفاده می کند به این معنای تصویر درون یک شکل نمایش داده می شود نه می تواند شباهت های

گرافیکی مختلفی اعمال شود . برای پیاده سازی آن ، وقتی از دستور plt.show() استفاده

می شود تصویر باقی می ماند .

کاربرد
⑤ بهترین روشی برای استفاده از OpenCV برای نمایش تصویر مناسب است

در حال حاضر استفاده از matplotlib برای نمایش های علمی مناسب تر و کاربردی تر است .