

گزارش کار پروژه دوم

هوش مصنوعی

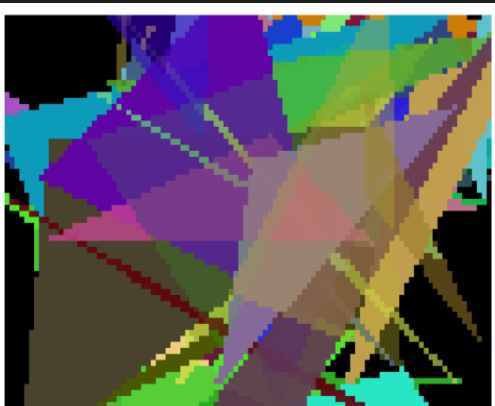
کسری کاشانی

810101490

- قسمت اول: الگوریتم ژنتیک

ابتدا روند تولید دو عکس moon و eagle با $\text{population} = 50$ و $\text{triangles_number} = 50$ در شکل های زیر در generation های 0 و 2000 و 7000 قابل مشاهده می باشند.

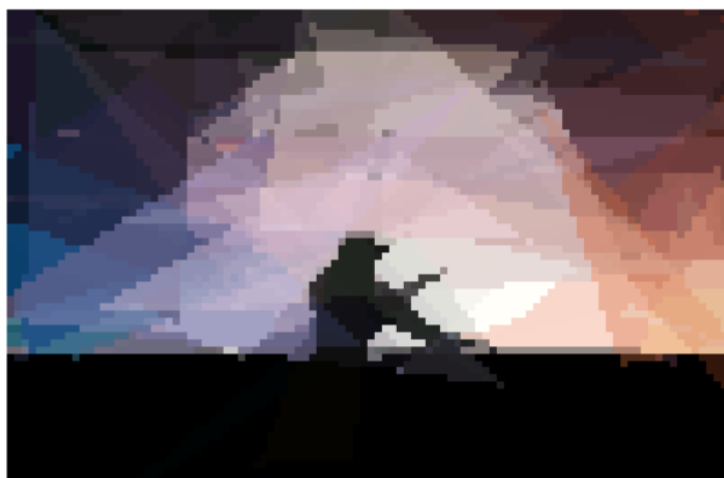
Fitness in Generation 0: mean: -3727.3280387543246, max: -2452.580561322568 min: -5077.494086889657



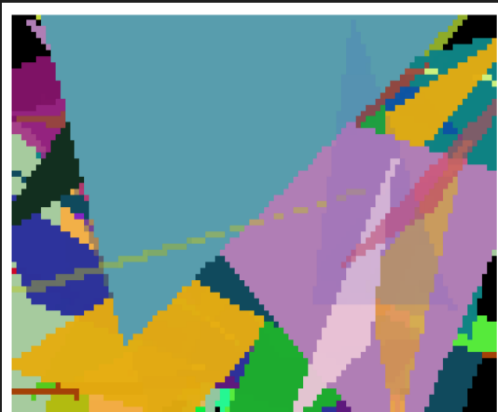
Fitness in Generation 2000: mean: -259.33617839292583, max: -259.23501730103806 min: -259.4144252210688



Fitness in Generation 7000: mean: -110.77053779315648, max: -110.76919646289889 min: -110.77056516724336



Fitness in Generation 0: mean: -3824.3408246059203, max: -2982.01593233718 min: -4794.215547866205



Fitness in Generation 2000: mean: -224.03810472895043, max: -222.2953171856978 min: -224.13273356401382

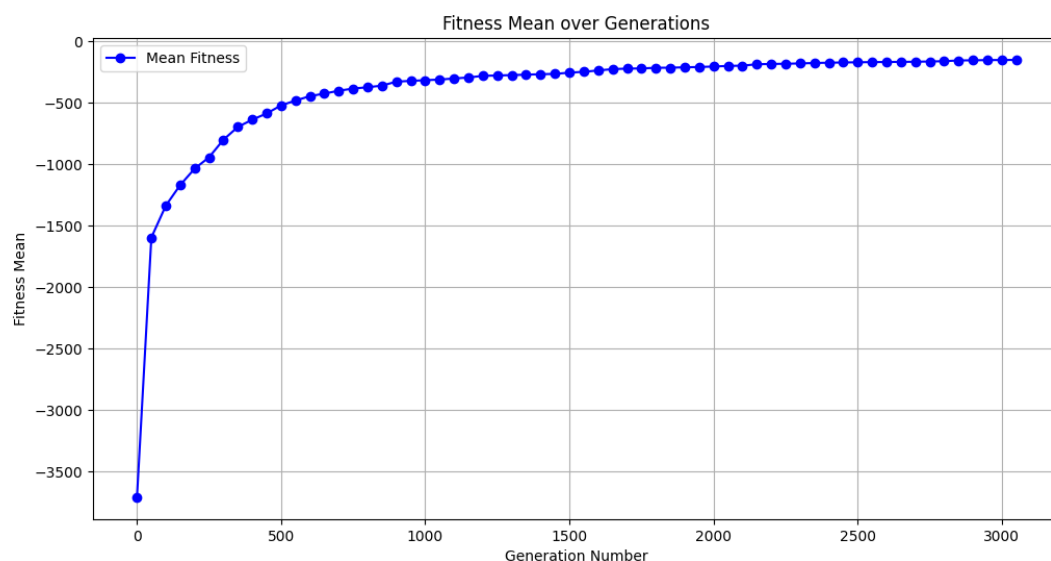


Fitness in Generation 7000: mean: -111.93959738562091, max: -111.91166474432909 min: -111.96985774702036



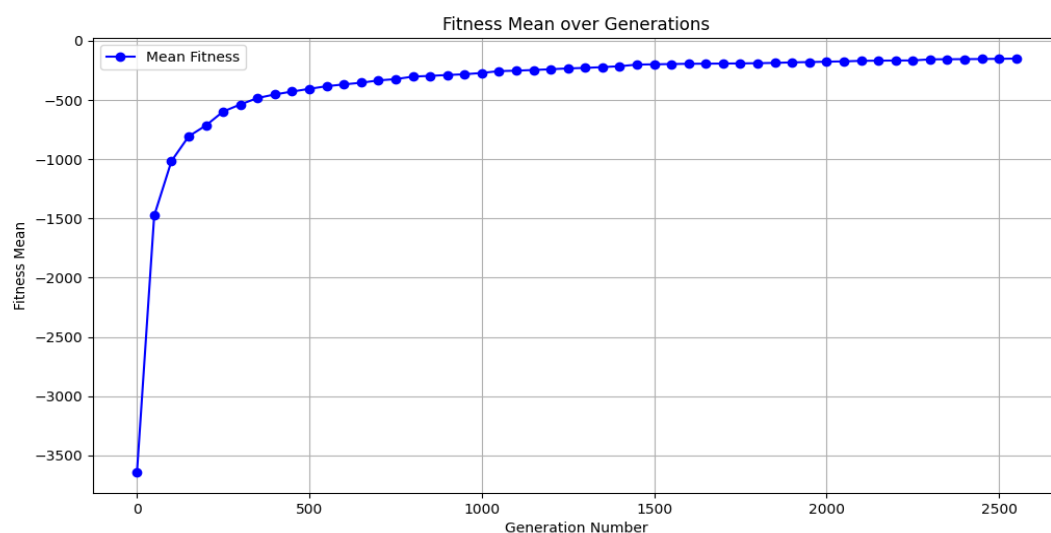
حال برای عکس moon نمودار های مقایسه ی روند افزایش fitness را در سه حالت با جمعیت های مختلف یعنی 50 و 70 و 100 بررسی می کنیم و می بینیم که در هر حالت پس از گذشتن چند نسل میانگین fitness به 150- می رسد.

با جمعیت 50:



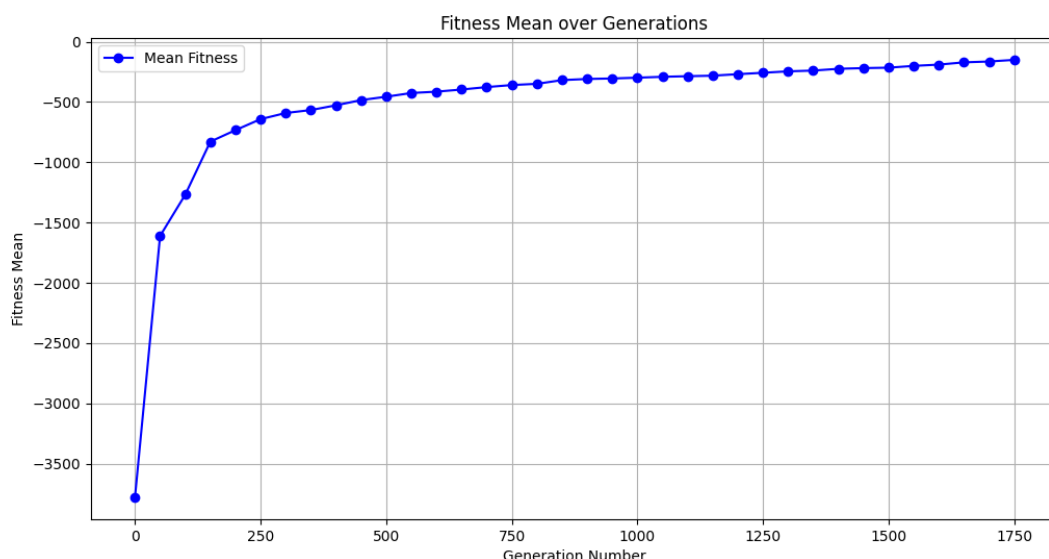
پس از گذشتن 3050 نسل به میانگین 150 fitness- رسیدیم.

با جمعیت 70:



پس از گذشتن 2550 نسل به میانگین 150 fitness- رسیدیم.

با جمعیت 100:



پس از گذشتن 1750 نسل به میانگین fitness -150 رسیدیم.

همانطور که از نمودار های مقایسه مشخص است، سرعت همگرا شدن fitness در کمترین جمعیت، کمترین است و نیاز به نسل های بیشتری برای رسیدن به آن fitness دارد. سپس با افزایش جمعیت، سرعت همگرا شدن fitness افزایش و تعداد نسل ها کاهش می یابد و بالاخره در بیشترین جمعیت، سرعت همگرا شدن fitness به بیشترین حالت خود می رسد و نسل های کمتری برای رسیدن به آن fitness تولید می شوند. در نتیجه، با افزایش تعداد جمعیت نسل ها، fitness سریع تر و در generation های کمتری همگرا می شود.

حال چند معیار را برای سنجش تفاوت دو عکس بررسی می کنیم. معیار اول می تواند میانگین خطای مطلق یا MAE باشد که همان میانگین قدرمطلق تفاوت پیکسل های دو عکس تولید شده و هدف می باشد. معیار دوم نیز می تواند مجموع مربع خطاها یا MSE باشد که همان مجموع مربعات تفاوت پیکسل های دو عکس تولید شده و هدف می باشد. معیار سوم هم می تواند Peak Signal to Noise Ratio یا PSNR باشد که همان مقایسه ی کیفیت و دقت عکس تولید شده در برابر عکس اصلی می باشد. در این پروژه، از معیار دوم یعنی MSE به عنوان یک معیار سازگار برای سنجش تفاوت دو عکس استفاده می کنیم. چرا که نحوه پیاده سازی و محاسبات این روش خیلی ساده و سریع است و کروموزوم ها سریع تر ارزیابی می شوند و سرعت کلی و کارایی برنامه نیز افزایش می یابد.

سوال 1) کروموزوم ها در این پروژه، مجموعه ای از مثلث ها هستند که هر مثلث یا ژن نیز دارای ویژگی هایی مانند رنگ و مختصات می باشد. از آنجا که هر کروموزوم شامل ژن هایش یعنی مثلث ها می باشد، ابتدا فضای حالت هر مثلث را بررسی می کنیم. بدین منظور تعداد تمام ترکیبات ممکن برای هر ویژگی را محاسبه می کنیم و در یکدیگر ضرب می کنیم تا فضای حالت یک مثلث محاسبه شود. رنگ هر مثلث دارای 4 کانال می باشد که هر کدام یک مقدار از 0 تا 255 می توانند بگیرند یعنی 256 حالت. پس برای هر مثلث 256^4 رنگ متفاوت وجود دارد. همچنین هر مثلث دارای 3 راس است که هر راس دارای مختصات (x,y) می باشد و هر مختصات نیز می تواند دارای یک مقدار از $(0,0)$ تا $(width-1,height-1)$ باشد یعنی $width \times height$ حالت. پس برای هر مثلث $(width \times height)^3$ مختصات متفاوت وجود دارد. در نتیجه فضای حالت هر مثلث یا هر ژن برابر با $(width \times height)^3 \times 256^4$ می باشد. حال اگر هر کروموزوم دارای n مثلث یا ژن باشد، فضای حالت هر کروموزوم برابر با $(width \times height)^3 \times 256^4$ خواهد بود.

سوال (2) در صورتی که crossover یا mutation در الگوریتم انتخاب های متنوعی نداشته باشد یا بیش از حد سختگیرانه انتخاب کند، ممکن است الگوریتم به یک پاسخ غیر بهینه همگرا شود و ژن ها پیشرفت نکنند. یکی از ایده ها، نخبه گرایی یا حفظ بهترین افراد یک جمعیت برای انتقال به نسل بعد می باشد. بدین صورت که چند کروموزوم اول هر جمعیت که fitness بالاتری دارند را به طور مستقیم و بدون crossover یا mutation به نسل بعدی منتقل می کنیم. لذا الگوریتم با سرعت بالاتری می تواند پاسخ های بهینه را تولید و پیدا کند و زمان همگرایی کاهش می یابد. ایده ی دیگر، استفاده از نرخ جهش تطبیقی است که در آن نرخ جهش یا mutation rate بر اساس وضعیت جمعیت آن نسل تغییر می کند و مشکل کاهش تنوع جمعیت با گذر زمان از بین می رود. در واقع باید در ابتدای الگوریتم این نرخ جهش را بالا در نظر بگیریم و با گذر زمان و همگرا شدن تقریبی الگوریتم، نرخ جهش را کاهش دهیم تا کروموزوم های بهینه در هر نسل باقی بمانند و تغییر نکنند.

سوال (3) چندین استراتژی برای انتخاب نسل بعدی وجود دارد. مثلا در استراتژی انتخاب چرخ رولت یا همان Roulette Wheel Selection، احتمال انتخاب هر جفت کروموزوم به نسبت شایستگی آن ها می باشد. یعنی کروموزوم های با شایستگی بیشتر دارای احتمال بالاتری برای انتخاب شدن در نسل بعدی دارند. این روش با اینکه معمولا سریع به همگرایی می رسد، اما ممکن است تنوع جمعیت در هر نسل از بین برود. یا مثلا در استراتژی انتخاب مبتنی بر رتبه یا همان Rank Based Selection، کروموزوم ها ابتدا بر اساس رتبه ی شایستگی خود مرتب می شوند و سپس به نسبت رتبه ی خود، احتمال انتخاب شدن در نسل بعدی را دارند. در این روش مشکل از بین رفتن تنوع جمعیت در هر نسل وجود ندارد.