Abstract

In the field of robotics it is still somewhat of a challenge to analyse objects and scenes in camera images that are far away from the robot. This might not be problematic for many stationary and/or slow moving platforms, but it is easy so see what issues may arise when the speed increases or timing in general gets critical.

Combating this problem can be done by, for example, changing the lens of the camera, which in turn hurts the field of view (FOV).
Another approach would be to employ a higher resolution camera. The problem here is that more bandwidth, aswell as higher computing power would be needed to transport and process those images. So every advance into more depth clarity comes with it's own tradeoffs.

Therefor a solution with a smaller cost is needed. For this goal, the "higher resolution" approach may still be a viable option by using AI upscaling. The idea is to use a supersampling model to "enhance" certain regions of interest in the source image and analyse them further. For the sake of evaluation this further analysis will be performed as monocular depth estimation.

Introduction

A person operating a motor vehicle always needs to be aware of their surroundings and the environment. They know that the car they saw coming on the background is not gone because the turning car in front of them is blocking it. This phenomena is called object permanency and is something the majority of humans learn growning up. That same concept is not hard for a computer to grasp either but what might be is the preamble of "detecting the car in the background". Robots rely heavily on images to orient themselves in their environments but this brings some limitations. There are some options to combat this, but they either come with their own tradeoffs or may not be aplicable in all situations.

Changing to a camera with a thicker lense will improve the zoom, but since the image size stayed the same the overall field of view (FOV) is decreased. This may affect close by detection and analysis. As a fix for the above method or a standalone approach it might be useful to change recording resolution to something greater. But also this approach has drawbacks as it required way more bandwith and computing power to process those larger images. Another fix to the lens problem might be varifocal lenses but they add in complexity/cost and might not be easily applicable in stereo camera setups since both lenses would need to be well synchronized.

A comparatively low cost solution arises in image superresolution implemented as an AI supersampling model. With this method no extra hardware equipment would be needed. There would still be an overhead though, but it can be held down by only supersampling specific regions of interest like the end of a road, intersections, entrances, etc.

Background

Monocular Depth Estimation

Depth sensing is traditionally done utilizing multiple lenses with a fixed and known distance and orientation. This concept is based on observations of nature and is also the way our eyes work. But apart from this, there are not many methods of "spatial vision". One of those other methods is monocular depth estimation where a depth image is to be guessed from a single perspective by a neural network. Initial works in this area of research are for example . There attempts were made to utilize Markov random fields and simpler versons of current concepts like DNN feature encoding. Through time many implementations and methods for optiomization where developed. Reletive recently proposed Global-Local Path Networks consisting of an encoder and decoder following a global path with skip connections running into "Selective Feature Fusion" (SFF) modules for local features. The general structure can be seen in figure 1.

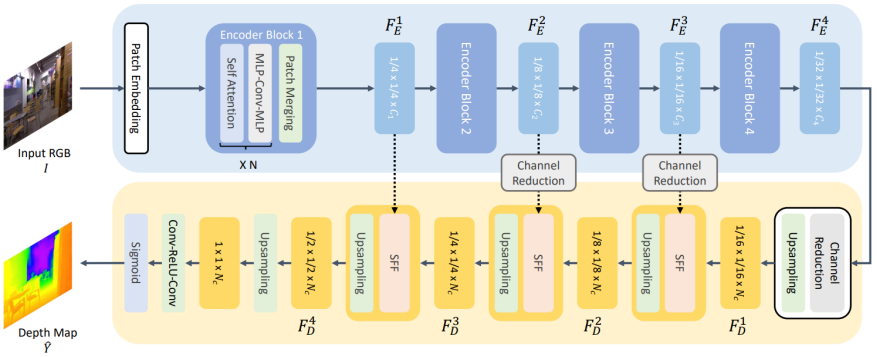

Figure 1: GLPDepth architecture

The Encoder consists of 4 (encoding) feature stages with an encoding block each. Between stages the dimensions get reduced by scales $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$. This in done to reduce computational overhead. The first 3 stages also host skip connections into the decoder wich will be talked about later.

The Decoder itself follows a similar but reversed structure as the encoder. It scales the dimensions back up by $\frac{1}{16}$, $\frac{1}{8}$, $\frac{1}{4}$ and $\frac{1}{2}$. Inbetween those upsampling steps are so called SFF modules wich merge global features with local features from the skip connections. Their specific construction is as depicted in figure 2.

According to the researchers, a big performance boost for GLPDepth was also a data augmentation method they called Vertical CutDepth. It is an offspring of the CutDepth method which cuts a region of interest (ROI) from the source image to create a "new" datapoint. The difference with Vertical CutDepth is, that the ROI only takes a random horizontal position and width but

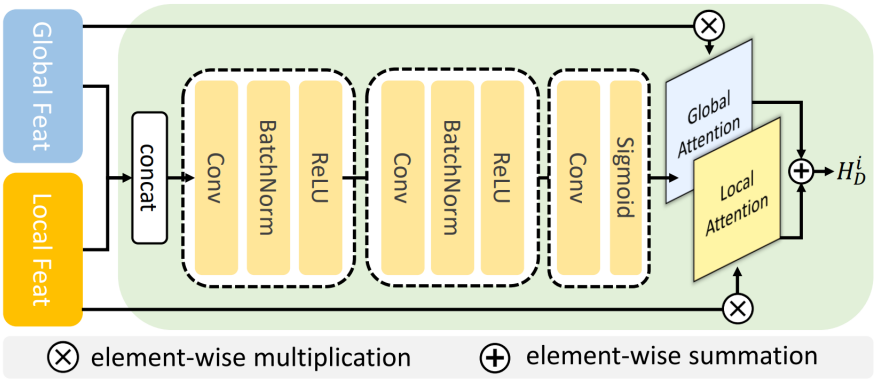

Figure 2: SFF architecture

always remains at $y = 0$ with the full height. This is due to findings in previous depth estimation networks which suggest that they focus more on veritcal then horizontal features.

Implementation

Experimental Evaluation

Conclusion