

# DYNAMIC HETEROGENEOUS SENSOR REGISTRATION FOR VEHICLE PERCEPTION VIA DEEP NEURAL NETWORKS

*Michael Giering, Vivek Venugopalan, Kishore Reddy*

United Technologies Research Center  
Decision Support and Machine Intelligence Group  
411 Silver Lane, East Hartford CT 06108

## ABSTRACT

When performing multi-modal fusion to perform an analytic task, spatio-temporal alignment of the incoming signals is often a key to the stability of all methods subsequently applied to the fused data. Lidar-Video systems like those many driverless cars are a common example of where keeping the Lidar and video channels registered to common physical features is important. We develop a deep learning technique that takes multiple channels of heterogeneous data, to test whether we can detect misalignment of the Lidar-video inputs. A number of variations were tested on the Ford LV driving test data set.

## 1. MOTIVATION

Navigation and situational awareness of optionally manned vehicles requires the integration of multiple sensing modalities such as LIDAR and video, but could just as easily be extended to Radar, SWIR and GPS. Spatio-temporal registration of information from multi-modal sensors is technically challenging in its own right. For many tasks such as pedestrian and other object detection tasks that make use of multiple sensors, decision support methods rest on the assumption of proper registration. Most approaches in LIDAR-video for instance, build separate vision and lidar feature extraction methods and then try to identify common anchor points in both. The ability to dynamically register information from the available data channels for perception related tasks can alleviate this achilles ankel of multi-modal sensor fusion.

Deep neural networks lend themselves in a seamless manner for data fusion on these types of problems. By building a single model on all information being collected at once, it has been shown that features generated on the fused information [1] can provide insight that neither input alone can. From a more applied perspective, it is possible to create such systems with far less overhead. The need for domain experts and hand-crafted feature design are lessened, thereby allowing more rapid prototyping and testing.

The trained nnets easily run within the real-time constraints of common frame rates and lidar data collection.

The generalization of autoregistration across multiple assets is clearly a path to be explored.

By including dynamic flow as input channels, we imbue the nnet with information on the dynamics observed across time steps.

## 2. PREVIOUS WORK

Need some references to define the state of the art

## 3. PROBLEM STATEMENT

need detailed description and citation regarding the ford data set need greater clarity on the optical flow method used.

Being able to detect and correct the misalignment among sensors of the same or different kinds is critical when operating on the fused information emanating from the sensors. For this work DCNN's were implemented for the detection of small spatial misalignments in Lidar and Video frames. The data was collected from a driverless car was chosen as the multi-modal fusion test case. LV is a common combination for providing perception capabilities to many types of ground and airborne robots including driverless cars [google, ford].

### 3.1. Ford LV data set and experimental setup

**Kishore -Detailed description of the ford data set [1], our test and training and the justifications for it. framerate. Vivek - a brief description of the hardware used.** As shown in diagram-n, we divided the data set into training and testing sections A and B. They were chosen in a manner that minimizes the likelihood of contamination between training and testing. Because of this, the direction of the lighting is source is never the same in the testing and training sets. This provides an additional measure of testing the generalizability of our models.

### 3.2. Preprocessing

At each video frame timestep, the inputs to our model consisted of C-channels of data with C ranging from 3-6 channels. Channels consisted of inputs that included greyscale and (R,G,B)-video channels, horizontal and vertical optical flow and Lidar depth information. Each channel was cropped to a uniform 800x??? pixels. Each time step has an 800 x ??? x C array of integer values.

These arrays were subdivided into  $p \times p \times C$  patches at a prescribed stride. For any experiment we can denote the preprocessing parameters

- R,G,B — Frame color channels.
- U,V — optical flow channels.
- L — lidar depth channel.
- C — number of input channels.
- p — patch size.
- s — stride.

For a given frame of size 800 x h there are approximately  $n = (800 \times h)/s$  patches (exact number?). The training and test sets had X and Y frames respectively, therefore the entire data set consists of  $N = n \times X$  inputs of the patch-size dimension.

Preprocessing is repeated O times, where O is the number of offset classes. For this work we used two setups. A 5 class, linearly distributed set of offsets and a 9 class elliptically distributed set of offsets. (see figure x) For each offset class, **Kishore explain how you generated the data.**

In order to accurately detect misalignment in the LV sensor data, we've assumed there needs to be a lower bound on the amount of information present in each channel. For this data set, L was the only channel with regions of low information. A preprocess step was to eliminate all patches corresponding to L data with variance  $\leq x$ . This leads to the elimination of the majority of foreground patches in the data set, reducing the size of the training set by **z pct KISHORE**

## 4. MODEL DESCRIPTION

**need to describe the parameters post-processing,classification metric for each patch,a table with common params for the experiments would help,voting scheme**

The model consists of a 4-layer ? CNN classifier *see image of network* that estimates the offset between the LV inputs at each time step. For each patch within a timestep, there are O variants with the LVF inputs offset by the pre-determined amounts. The CNN outputs to a softmax layer, thereby providing an offset classification value for each patch of the frame. figure x: In the 5 class example we color each

patch of the frame with a color corresponding to the predicted class.

For each frame a simple voting scheme is used to aggregate the patch level offset predictions to frame level predictions. A sample histogram of the patch level predictions is show in figure x.

### 4.1. optical flow

*kishore, please discuss the motivation to include dynamics, how we performed it and how we'd need to do it if running in real time. this is where we can point ot proof that it improves prediction.*

## 5. EXPERIMENTS AND POST-PROCESSING

*Need a complete list of the experiments run images to visualize the frame level results please place any confusion matrices and your comments on what you think the results say. feel free to suggest any tables or other visuals to include.*

### 5.1. 5 class tests

In our initial tests, the linearly distributed set of 5 offsets of the LV data were performed. Table 1 lists the inputs and CNN parameters explored ranked in the order of increasing accuracy (**define accuracy and other cm metrics**), **include training vs test error and conf mats if room allows.**

As can be seen ...

### 5.2. 9 class tests

The subsequent tests were designed to understand whether the simple linear displacement model of the 5-class test could be generalized to a model capable of discriminating multiple directions and displacement magnitude. To acheive this 8 positions were chosen on an ellipse along with it's center **describe the parabola**. LV was offset in a manner similar to the 5 class test. Nine training and test sets were generated and an identicle patch level CNN was constructed differing only in the 9 class softmax output layer.

Table 2 lists the inputs and CNN parameters explored ranked in the order of increasing accuracy (**define accuracy and other cm metrics**), **include training vs test error and conf mats if room allows.**

**Discussion: what results confirmed expectations or surprised us (grey scale). Can we confidently say optical flow improves prediction.**

## 6. CONCLUSION AND FUTURE WORK

We did it. We're great.

future: implement a method that doesn't require ground truth and also generalizes easily to a wide array of sensors.

Test it on data collected from airborne platforms that are noisier and have more degrees of freedom.

## **7. REFERENCES**

populate the papers to be cited in the folder and if possible the bib file

## **8. REFERENCES**