MULTI-MODAL SENSOR REGISTRATION FOR VEHICLE PERCEPTION VIA DEEP NEURAL NETWORKS

Michael Giering, Kishore Reddy, Vivek Venugopalan
Decision Support & Machine Intelligence Group
United Technologies Research Center
E. Hartford, CT 06060, USA
Email: gierinmj, kkreddy, venugov@utrc.utc.com

ABSTRACT

The ability to simlutaneously leverage multiple modes of sensor information is critical for perception of an automated vehicle's physical surroundings. Spatio-temporal alignment of registration of the incoming information is often a prerequisist to analyzing the fused data. The persistance and reliability of multi-modal registration is therefore the key to the stability of decision support systems ingesting the fused information. Lidar-Video systems like on 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 method that takes multiple channels of heterogeneous data, to detect the misalignment of the Lidar-video inputs. A number of variations were tested on the Ford LV driving test data set and will be discussed. To the best of our knowledge the use of multimodal deep convolutional neural networks for dynamic real-time LIDAR/video registration has not been presented.

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 other modalities including Radar, SWIR and GPS. Spatio-temporal registration of information from multimodal sensors is technically challenging in its own right. For many tasks such as pedestrian and object detection tasks that make use of multiple sensors, decision support methods rest on the assumption of proper registration. Most approaches Bodensteiner & Arens (2012) in LIDAR-video for instance, build separate vision and lidar feature extraction methods and identify common anchor points in both. Alternatively, by generating a single feature set on Lidar, Video and optical flow, it enables the system to to capture mutual information among modalities more efficiently. The ability to dynamically register information from the available data channels for perception related tasks can alleviate the need for anchor points *between* sensor modalities. We see auto-registration as a prerequisit need for operating on multi-modal information with confidence.

Deep neural networks lend themselves in a seamless manner for data fusion on time series data. It has been shown [Ng multimodal] for some challenges in which the modalities share significant mutual information, the features generated on the fused information can provide insight that neither input alone can. In effect the ML version of, ”the whole is greater than the sum of it's parts”.

Speed constraints of real time navigation also constrain model selection. The trained nnets easily run within the real-time constraints of common frame rates and lidar data collection.

From an applied research 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 generalization of autoregistration across multiple assets is clearly a path to be explored.

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

2 Previous Work

Kishore here

A great amount has been published on various multimodal fusion methods. The most common approaches taken generate features of interest in each modality seperately and create a decision support mechanism that aggregates features across modalities. If spatial alignment is required across modalities, as it is for LiDAR/ video such filter methods Thrun (2011) are required to ensure proper inter-modal registration. These filter methods for leveraging 3D LiDAR and 2D images are often geometric in nature and make use of projections between the different data spaces.

The use of deep neural networks to analyze multimodal sensor inputs has increased sharply in the last couple of years, including audio/video Ngiam et al. (2011) Kim et al. (2013), image/text Srivastava & Salakhutdinov (2012), image/depth Lenz et al. (2013) and Lidar/video. To the best of our knowledge the use of multimodal deep neural networks for dynamic real time LIDAR/video registration has not been presented.

A common question often arises in data fusion methods, which is ”at what level should features from the differing sensor streams be brought together?” Most similar to the more traditional data fusion methods is to train DNN's independently on sensor modalities and then use the high-level outputs of those networks as inputs to a subsequent DNN. This is analogous to the earlier example of learning 3D/2D features and subsequently identifying common geometric features.

It is possible however to apply DNN's with a more agnostic view enabling a unified set of features to be learned across multimodal data. In these cases the input channels aren't differentiated. Unsupervised methods in particular applying DBM's for learning such joint representations have been successful.

DCNN's enable a similar agnostic approach to input channels. A significant difference is that target data is required to train them as classifiers. *find examples in the literature*. This is the approach chosen by us for automating the registration of LiDAR/video/optical-flow, in which we are combining 1D/3D/2D data representations to learn a unified model across as many as 6D.

3 Problem Statement

Being able to detect and correct the misalignment (registration, calibration) among sensors of the same or different kinds is critical when operating on the fused information emanating from them. For this work Deep Convolutional Neural Networks (DCNN) were implemented for the detection of small spatial misalignments in LIDAR and Video frames. The data collected from a driverless car was chosen as the multi-modal fusion test case. LIDAR-Video is a common combination for providing perception capabilities to many types of ground and airborne platforms including driverless cars Thrun (2011).

3.1 Ford LIDAR-Video Dataset and Experimental Setup

Figure 1: Left: The modified Ford F-250 pickup truck. Right: Sample image from front facing camera and green dots indicate the region of lidar data.

The FORD LIDAR-Video dataset Pandey et al. (2011) is collected by an autonomous Ford F-250 vehicle integrated with the following perception and navigation sensors as shown in Figure 1:

- Velodyne HDL-64E LIDAR with two blocks of lasers spinning at 10 Hz and a maximum range of 120m.
- Point Grey Ladybug3 omnidirectional camera system with six 2-Megapixel cameras collecting video data at 8fps with 1600×1600 resolution.
- Two Riegl LMS-Q120 LIDAR sensors installed in the front of the vehicle generating range and intensity data when the laser sweeps its 80°field of view (FOV).
- Applanix POS-LV420 INS with Trimble GPS system providing the 6 degrees of freedom (DOF) estimates at 100 Hz.
- Xsens MTi-G sensor consisting of accelerometer, gyroscope, magnetometer, integrated GPS receiver, static pressure sensor and temperature sensor. It measures the GPS coordinates of the vehicle and also provides the 3D velocity and 3D rate of turn.

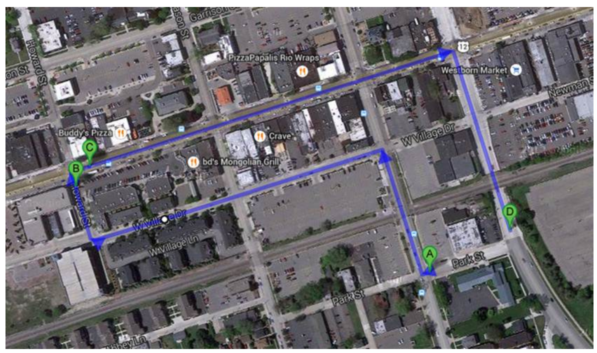

Figure 2: Training (A to B) and testing (C to D) tracks in the downtown Dearborn Michigan.

This dataset is generated by the vehicle while driving in and around the Ford research campus and downtown Michigan. The data includes feature rich downtown areas as well as featureless empty parking lots. As shown in Figure 2, we divided the data set into training and testing sections A to B and C to D respectively. They were chosen in a manner that minimizes the likelihood of contamination between training and testing. Because of this, the direction of the light source is never the same in the testing and training sets.

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 components of optical flow and Lidar depth information. Each channel was cropped to a uniform 800x256 pixels. Each time step has an 800 x 256 x C array of integer values.

These arrays were subdivided into p x p x C patches at a prescribed stride as shown in Figure 3. For any experiment we can denote the preprocessing parameters

- R,G,B — Frame color channels.
- W — Grey color channel.
- 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 x h)/s patches (exact number?). The training and test sets had X and Y frames respectively, therefore the entire training data set consists of N = n x 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 as shown in Figure 6. For each offset class, the video (R,G,B) and optical flow (U,V) channels are kept static and the depth (L) channel from the lidar is moved by the offset simulating a misalignment between the video and the lidar sensors. At a given timestep, all n patches extracted from the C-channels for different offsets are grouped in to different classes for training purposes. The information from each channel for a given patch location are vectorised into a d=p x p x C dimension vector resulting in a nxd dimension matrix for each offset class at a given timestep.

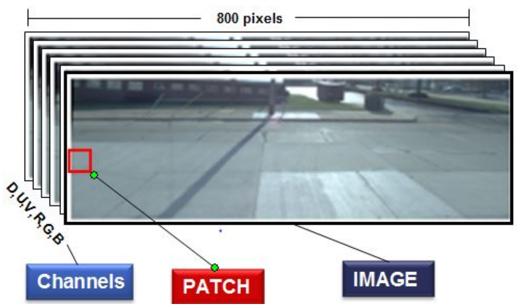

Figure 3: At each time step the channels were sampled at the noted patch size at a fixed stride

In order to accurately detect misalignment in the Lidar and Video 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 less than a threshold (<x). This leads to the elimination of the majority of foreground patches in the data set, reducing the size of the training set by approximately 80%. In the testing stage, the same threshold x is used to not perform classification on patches with less information.

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

Our models for auto-registration are DCNN's trained to classify the current misalignment of the LiDAR/video data streams into one of a predefined set of offsets. DCNN's are probably the most successful deep learning model to date on fielded applications. The fact that the algorithm shares weights in the training phase, results in fewer model parameters and more efficient training. DCNN's are particularly useful for problems in which local structure is important, such as object recognition in images and temporal information for voice recognition. The alternating steps of convolution and pooling **(as depicted in figure X)** generates features at multiple scales which in turn imbues DCNN's with scale invariant characteristics.

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 predetermined 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.

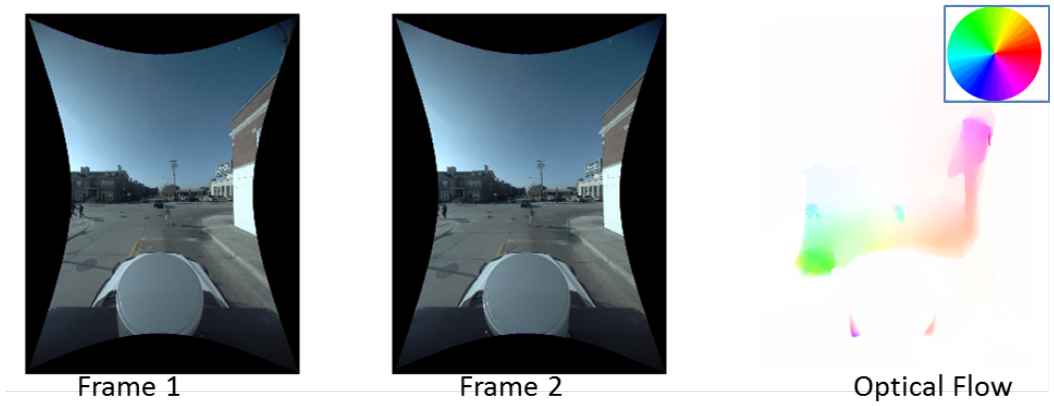

Figure 4: Optical flow: Hue indicates orientation and saturation indicates magnitude

4.1 Optical Flow

In the area of navigation of mobile robots, optical flow has been widely used to estimate egomotion [Prazdny (1980)], depth maps [Shahraray & Brown (1988)], reconstruct dynamic 3D scene depth [Yang et al. (2012)], and segment moving object [Chien et al. (2002)]. In our work we use optical flow as an additional channel derived from two consecutive frames to capture the dynamics. Optical flow gives an estimate of velocity at each pixel given two consecutive frames. Chase et al. (2008) demonstrated that optical flow can be performed in real-time on FPGA and GPU architectures. In our work, we used the algorithm described in [Liu (2009)] for computing optical flow. Figure 4 shows an example of the optical flow obtained using two frames from Ford LV dataset.

5 Experiments and Post-processing

The NVIDIA Kepler series K40 GPUs [NVIDIA Inc. (2012)] are very FLOPS/Watt efficient and are being used to drive real-time image processing capabilities [Venugopal & Kannan (2013)]. These GPUs consist of 2880 cores with 12 GB of on-board device memory (RAM). Deep Learning applications have been targeted on GPUs previously in [Krizhevsky et al. (2012)] and these implementations are memory bound. A GPU with higher memory capacity is excellent for these experiments due to the number of channels that are stacked and provided as the input to the DCNN. The LIDAR-Video dataset is augmented with the optical flow information resulting in 6 channels which means that the input to the DCNN is $32 \times 32 \times 6$ per patch. Figure 5 shows the setup used for the DCNN implementation on the GPU, where C defines the number of channels.

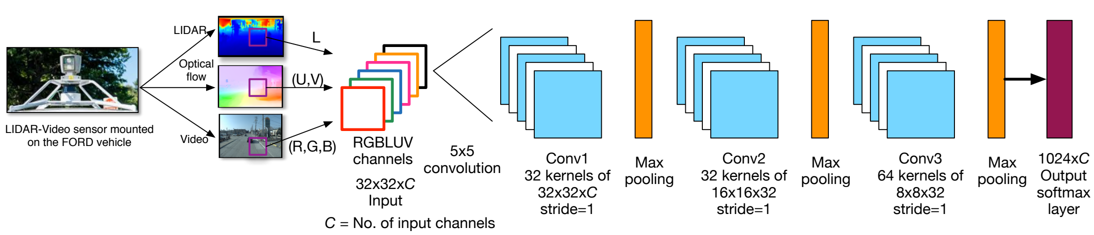

Figure 5: Experimental setup of the LIDAR-Video DCNN with 5×5 convolution

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 achieve 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 identical patch level CNN was constructed differing only in the 9 class softmax output layer.

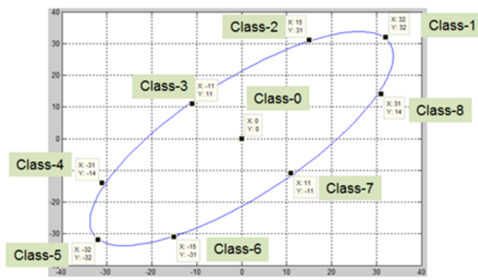

Figure 6: Placeholder: Visualization of the LiDAR / video channel relative offsets

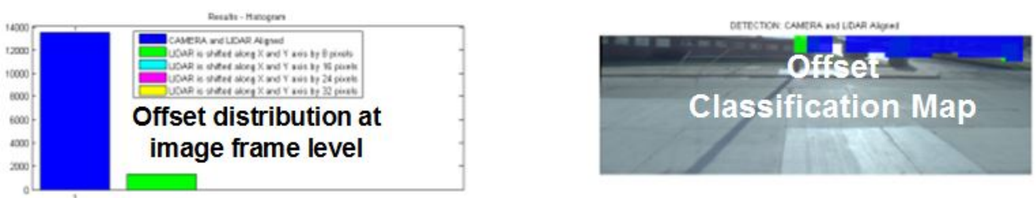

Figure 7: Placeholder: Voting

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 CONCLUSIONS AND FUTURE WORK

We did it. We're great.

The next step in taking this work forward is to complete our development of a deep auto-registration method for ground and airborn platforms requiring no apriori calibration ground truth. Our airborne applications in particular present noiser data with an increased number of degrees of freedom. The extension of these methods to simultaneously register information across multiple platforms and larger numbers of modalities will provide interesting challenges that we look forward to working on.

References

Bodensteiner, Christoph and Arens, Michael. Real-time 2D Video 3D LiDAR Registration. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pp. 2206–2209. IEEE, 2012.

Chase, J., Nelson, B., Bodily, J., Wei, Zhaoyi, and Lee, Dah-Jye. Real-Time Optical Flow Calculations on FPGA and GPU Architectures: A Comparison Study. In *Field-Programmable Custom Computing Machines, 2008. FCCM '08. 16th International Symposium on*, pp. 173–182, April 2008. doi: 10.1109/FCCM.2008.24.

Chien, Shao-Yi, Ma, Shyh-Yih, and Chen, Liang-Gee. Efficient moving object segmentation algorithm using background registration technique. *Circuits and Systems for Video Technology, IEEE Transactions on*, 12(7):577–586, Jul 2002. ISSN 1051-8215. doi: 10.1109/TCSVT.2002.800516.

Kim, Yelin, Lee, Honglak, and Provost, Emily Mower. Deep Learning for Robust Feature Generation in Audiovisual Emotion Recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 3687–3691. IEEE, 2013.

Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet Classification With Deep Convolutional Neural Networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Lenz, Ian, Lee, Honglak, and Saxena, Ashutosh. Deep Learning for Detecting Robotic Grasps. *arXiv preprint arXiv:1301.3592*, 2013.

Liu, Ce. *Beyond Pixels: Exploring New Representations and Applications for Motion Analysis*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2009. AAI0822221.

Ngiam, Jiquan, Khosla, Aditya, Kim, Mingyu, Nam, Juhan, Lee, Honglak, and Ng, Andrew Y. Multimodal Deep Learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 689–696, 2011.

NVIDIA Inc. NVIDIA's Next Generation CUDA Compute Architecture: Kepler TM GK110. Whitepaper, May 2012.

Pandey, Gaurav, McBride, James R, and Eustice, Ryan M. Ford Campus Vision And Lidar Data Set. *The International Journal of Robotics Research*, 30(13):1543–1552, 2011.

Prazdny, K. Egomotion and relative depth map from optical flow. *Biological Cybernetics*, 36(2): 87–102, 1980. ISSN 0340-1200. doi: 10.1007/BF00361077. URL `http://dx.doi.org/10.1007/BF00361077`.

Shahraray, B. and Brown, M.K. Robust depth estimation from optical flow. In *Computer Vision., Second International Conference on*, pp. 641–650, Dec 1988. doi: 10.1109/CCV.1988.590045.

Srivastava, Nitish and Salakhutdinov, Ruslan. Multimodal Learning With Deep Boltzmann Machines. In *Advances in neural information processing systems*, pp. 2222–2230, 2012.

Thrun, Sebastian. Google's driverless car. *Ted Talk, Ed*, 2011.

Venugopal, Vivek and Kannan, Suresh. Accelerating real-time lidar data processing using gpus. In *Circuits and Systems (MWSCAS), 2013 IEEE 56th International Midwest Symposium on*, pp. 1168–1171, August 2013. doi: 10.1109/MWSCAS.2013.6674861.

Yang, Y., Liu, Q., Ji, R., and Gao, Y. Dynamic 3D Scene Depth Reconstruction via Optical Flow Field Rectification. *PLoS ONE*, 7:47041, November 2012. doi: 10.1371/journal.pone.0047041.