NEURO NEUER

ABSTRACT

This paper reports an approach toward engineering, an evolving robot behaviour for a role of a goalkeeper “Neuro Neuer”: he will keep an eye on the ball (eDVS) and try to keep his net safe. Based on the data received from the (eDVS), he will track the ball position and velocity. From this data set, he will use a neural network to predict, in which direction he needs to move.

We will take advantage of the event based sensor to extract the position (cluster detection) of the ball. The position (x,y) and velocity (dx,dy) vectors will be fed as inputs into the neural network and the servo position in degrees will be our output. The neural network will be trained on the data several times. Afterwards the network will be able to predict the desired positon of the servo.

INTRODUCTION

What we used:

* Wooden frame (about 40cm x 80cm, height adaptable)
* Servos
* Arduino Microcontroller
* DVS

DVS

The notion of a “frame” of video data is embedded in machine vision. High speed frame-based vision is expensive because it is based on a series of pictures taken at a constant rate. The pixels are sampled repetitively even if their values are unchanged. Successive frames contain enormous amounts of redundant information, wasting memory access, RAM, disk space, energy, computational power and time. In addition, each frame imposes the same exposure time on every pixel, making it difficult to deal with scenes containing very dark and very bright regions. Short-latency vision problems require high frame rate and produce massive amount of input data. At high frame rate, few CPU instructions are available for processing each pixel.  This high data rate, besides requiring specialized computer interfaces and cabling, makes it expensive in terms of power to deal with the data, especially in real time or embedded devices.

By contrast, in the camera used for this paper, data are generated and transmitted asynchronously only from pixels with changing brightness. Infact, it solves these problems by using patented technology that works like your own retina. Instead of wastefully sending entire images at fixed frame rates, only the local pixel-level changes caused by movement in a scene are transmitted – at the time they occur. The result is a stream of events at microsecond time resolution, equivalent to or better than conventional high-speed vision sensors running at thousands of frames per second. Power, data storage and computational requirements are also drastically reduced, and sensor dynamic range is increased by orders of magnitude due to the local processing.

In a situation where the camera is fixed and the illumination is not varying only moving objects generate events. This situation reduces the delay compared to waiting for and processing an entire frame.

Like an abstraction of some classes of retinal ganglion cell spikes seen in biology, each event that is output from the DVS indicates that the log intensity at a pixel has changed by an amount *T* since the last event. *T* is a global event threshold which is typically set to about 15% contrast in this goalie robot application. In contrast to biology, the serial data path used requires the events to carry address information of what pixels number has changed. The address encodes the positive or negative brightness changes (ON or OFF) with one bit and the rest of the bits encode the row and column addresses of the triggering pixel.

🡪 how we used and picture

SERVOS

Servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors. It uses position feedback to control its motion and final position. The input to its control is some signal, either analogue or digital.

 Inside there is a simple set-up: a small DC motor, potentiometer and a control circuit. As the motor rotates, the potentiometer's resistance changes, so the control circuit can precisely regulate how much movement there is and in which direction. When the shaft of the motor is at the desired position, power supplied to the motor is stopped. If not, the motor is turned in the appropriate direction. The desired position is sent via electrical pulses through the signal wire.

Servos are controlled by sending an electrical pulse of variable width, or pulse width modulation (PWM), through the control wire. There is a minimum pulse, a maximum pulse, and a repetition rate. A servo motor can usually only turn 90 degrees in either direction for a total of 180 degree movement. The servo motor expects to see a pulse every 20 milliseconds (ms) and the length of the pulse will determine how far the motor turns.

🡪 how we used and picture

NEURAL NETWORK

An artificial neural network is an information processing paradigm that is inspired by biological nervous system. It is composed of a large number og highly interconnected processing elements called neurons.

Advantages:

- a neural network can perform tasks that a linear program cannot

- when an element of the neural network fails, it can continue without any problem by their parallel nature

- a neural network learns and does not need to be reprogrammed

- it can be implemented in any application

- it can be implemented without any problem

Disadvantages:

- the neural network needs training to operate

- the architecture of a neural network is different from the architecture of microprocessors therefore needs to be emulated

- Requires high processing time for large neural networks

There are essentially three steps to model a neural network:

1. Building the network

2. Training the network

3. Testing the network

1. To build the network, you need to specify the number of hidden layers, neuron in each layer, transfer function, weight/bias learning function, and performance function.
2. The training data set consists of input signals assigned with corresponding target (desired output). The neural network is then trained using one of the supervised learning algorithms, which uses data to adjust the network’s weights and thresholds so as minimize the error in its predictions on the training set. If the network is properly trained, it has then learned to model the unknown function that relates the input variables to the output variables, and can subsequently be used to make predictions where the output is not known.
3. The last step is testing the performance of the developed model, so many samples are given to the network for the testing and evaluate the performance.

RESULTS AND CONCLUSION

🡪 small part of the main code in matlab

Bibliography

1 Robotic Goalie with..

2 Engineering goalkeeper..