**Greeting**

* Good morning, dear members of the examination committee (and dear audience).
* My name is Martin Beneš.
* Welcome to my bachelor thesis defense.

**Assignment**

* Study the PIR sensors and recognition/classification.
* Design the theoretical system, using the PIR sensor data, scanning the situation, possibly the count of people.
  + Predefined fuzzy logic system
  + Artificial learning system
* Implement the described algorithm.
  + Verify its functionality on real situation.
* Sum up the result.

**Design**

* **Hardware**
  + Sensor device (B+B Sensors:PIR STD, NodeMCU)
    - ESP8266 (WiFi), mDNS, HTTP
  + Classification server (software implemented)
* **Software**
  + Sensor device (C++/Arduino)
    - Sensing, homogenous segmentation
  + Monitor
    - Python3 = easy prototyping
      * NumPy, SciPy, scikit
      * MatPlotLib, PySerial
    - Presentation, (fusion)

**Feature extraction**

* CWT = edge function
* Segment borders (extremes), segmentation
* Artefacts
  + Merging the segments using fuzzy system
* Features
  + Mean
  + Variance
  + Scale of interpolated line
  + Length of artefact

**Classifier**

* Based on linear regression
  + Multiple classifiers, each determining one attribute
    - Distance
    - Orientation
    - Side
    - ...
* Spacial model of sensed area = matrix of fuzzy values
  + Formula for the index value from the classifier outputs
    - Index has unique vector of coefficients for each classifier
  + Indices do not separate the space homogenously

**Training**

* Manual labelling
  + Reference of synchronized video frame
* Rather inaccurate
* Definitely a room for improvement

**Postprocessing**

* **Localization** = cluster analysis
  + K-means **vs.** Partitioning around medoids
* **Counting the people** = minimal within-cluster sum of squares

**Implementation**

* **Sensor device**
  + Sensor = PIR STD (*B+B Sensors*)
  + MCU = NodeMCU
    - ESP8266 = WiFi
    - mDNS and HTTP
    - Programmed in C++/Arduino
* **Classification server**
  + Monitor (software) = Python
    - Number of libraries = easy prototyping
      * NumPy, SciPy, scikit
      * MatPlotLib
      * PySerial
  + Linux/Bash used as well

**Results**

* Posterior probability (percentage) of partial classifiers
* Evaluation:
  + How many positive samples were classified positive?
* Possible improvements to the future
  + Change the labelling method
  + Usage of multiple sensors
    - Fusion described theoretically, but not implemented