Maximilian Krüger
Rolf P. Würtz

Institut für Neuroinformatik
Ruhr-Universität Bochum, Germany
[maximilian.krueger,rolf.wuertz]@neuroinformatik.rub.de
http://www.neuroinformatik.ruhr-uni-bochum.de/

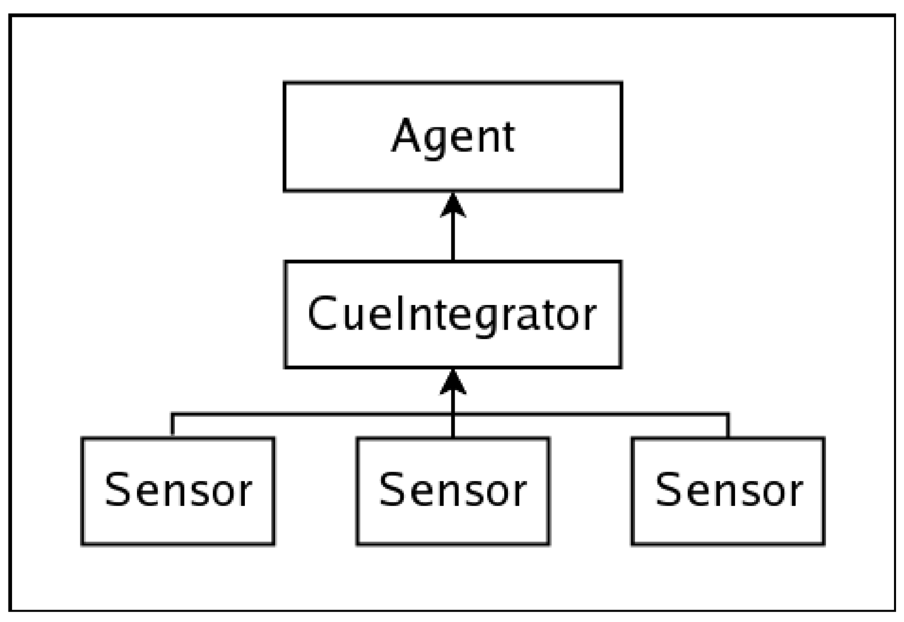
SPP 1183

Self-organized Evaluation of Dynamic Hand Gestures for Sign Language Recognition

Overview

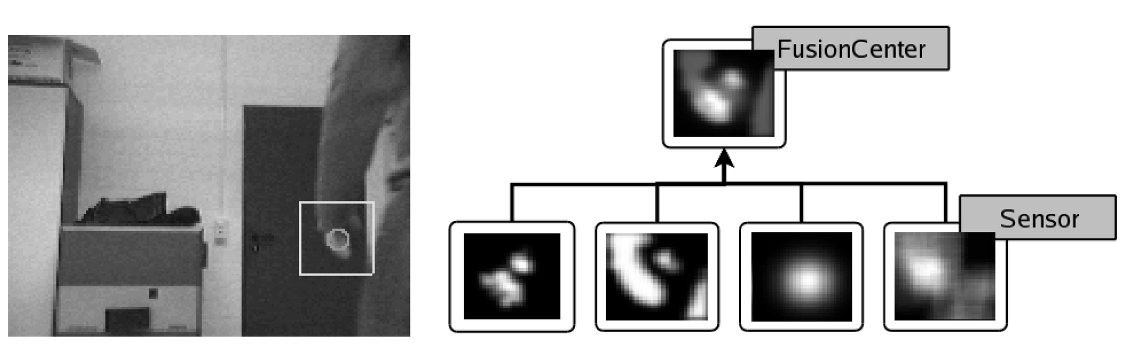
- Self-organization needs to be extended from the creation of static ordered structures to the organization of running processes.
- Classification of hand trajectories is a good example problem for that.
- Self-organization is here used to cope with time warping.
- The resulting system is applicable to human-machine interaction.

Agent architecture

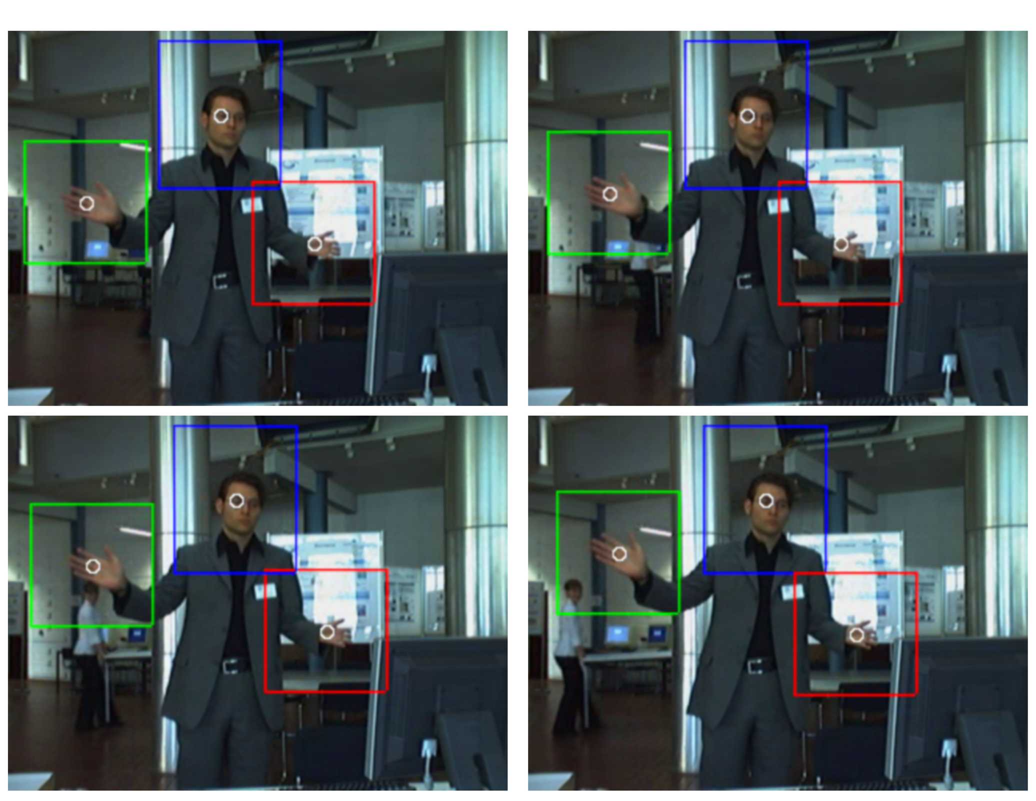


An agent is based on three modules provided with as much flexibility as possible. The interface to the environment and the communication are included in the agent class. Then there is one cueIntegrator, the module that integrates and interprets the information provided by the sensors.

Visual Tracking

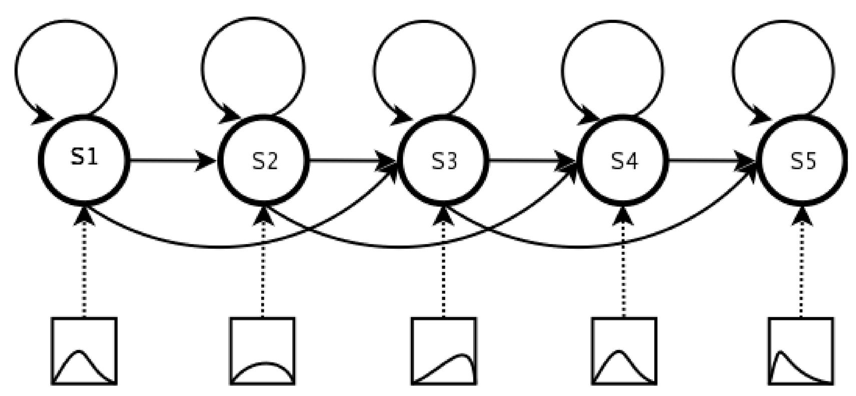


Tracking agent in use, on the left we see the tracking result marked with the circle. The rectangle shows the border of the agent's search region. On the right we see the similarity maps created by the different sensors, from left to right: color, motion, motion prediction and pixel template. The fusion center shows the result of the information integration.



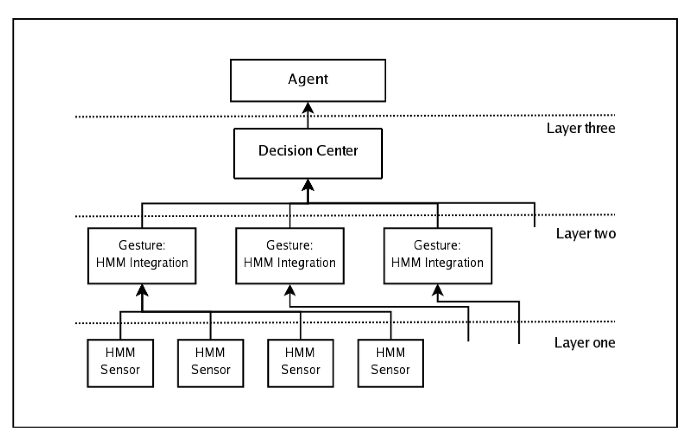
In this tracking sequence head and hands were found. The identity of the objects is visualized by the gray value of the rectangles, which delineate the search region of each tracking agent. Moving skin color in the background is ignored.

Modified Hidden Markov Model



HMM with the left-right (Bakis) topology, typically used in gesture and speech recognition. The solid lines denote the transition probabilities, the dotted line connects a continuous observation distribution to the belonging state (circle).
We have added a self-organizing aspect to this by evaluating the numbers of circle iterations and punishing the recognition value, respectively.

The HMM Recognition Agent



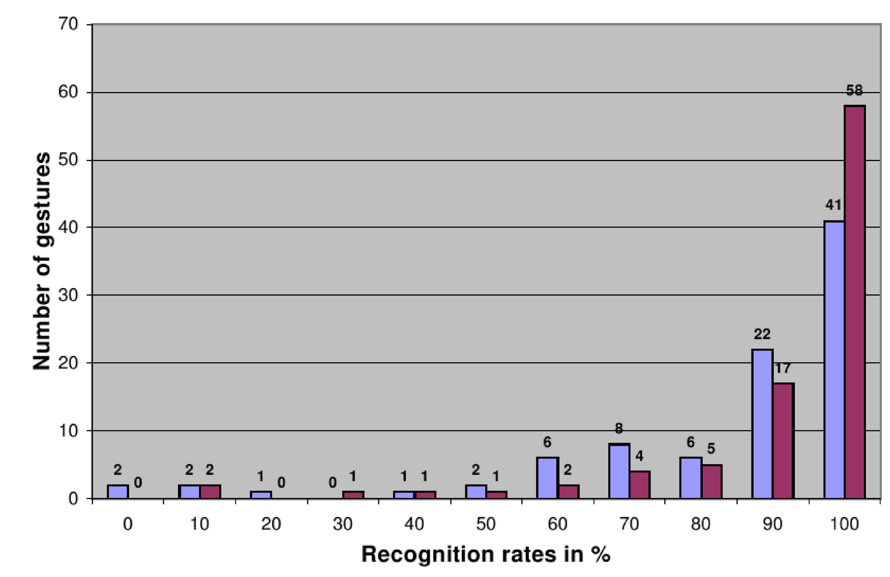
The Algorithm

while *not at end of gesture sequence* **do**
  `/* ******************************************** */`
  `/* Layer one:  HMM sensor */`
  `/* ******************************************** */`
  foreach *HMM sensor* **do**
    calculate observation probabilities;
  end
  `/* ******************************************** */`
  `/* Layer two:  HMM Integration unit */`
  `/* ******************************************** */`
  foreach *HMM Integration unit* **do**
    compute ϱ to fuse the information of position, texture and contour;
    calculate the actual quality Q_a;
    update the overall quality Q_g;
    control the activation using Q_g, ξ_{start} and ξ_{stop} ;
  end
  `/* ******************************************** */`
  `/* Layer three:  Decision Center */`
  `/* ******************************************** */`
  if $\exists$ *HMM Integration unit that reached its* ζ_{min} **then**
    choose HMM Integration unit with highest Q_g as current winner;
  end
  if $\zeta_{winner} == 1$ **then**
    reset all HMM Integration units;
  end
  else
    `/* inhibit all gestures */`
    search for the maximal quality Q_{max};
    foreach *HMM Integration unit* **do**
      subtract $Q_{max} from\ Q_g$;
    end
  end
end

Result: last winner will be chosen as recognized gesture.

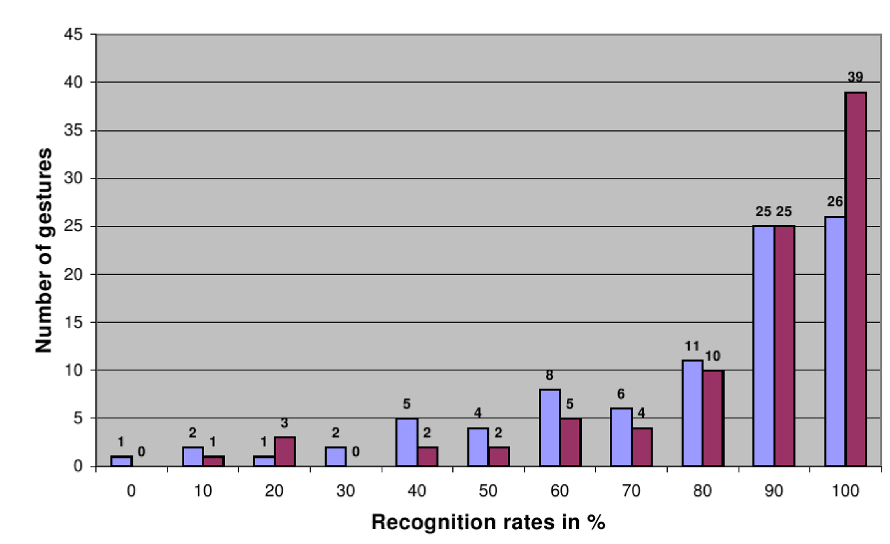
The recognition is hierarchically organized using three layers. The characteristic of each layer is its information integration. Layer one, the HMM Sensor compares the received observation with its observation probability function. Layer two comprise the HMM Integration unit of each learned gesture and integrates the information received from layer one. The top layer compares the results from the HMM Integration units. The Decision Center determines the most probable gesture and manages the inhibition

Results I



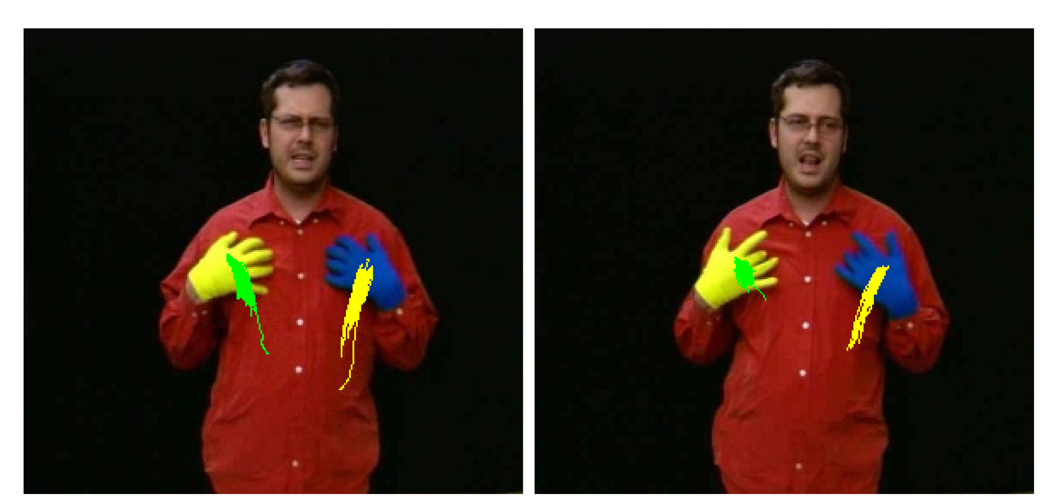
Recognition started with the beginning of the gesture. The histogram shows the result for the experiment using only position and contour information in light color and in dark color the result when integrating position, contour and texture information.

Results II



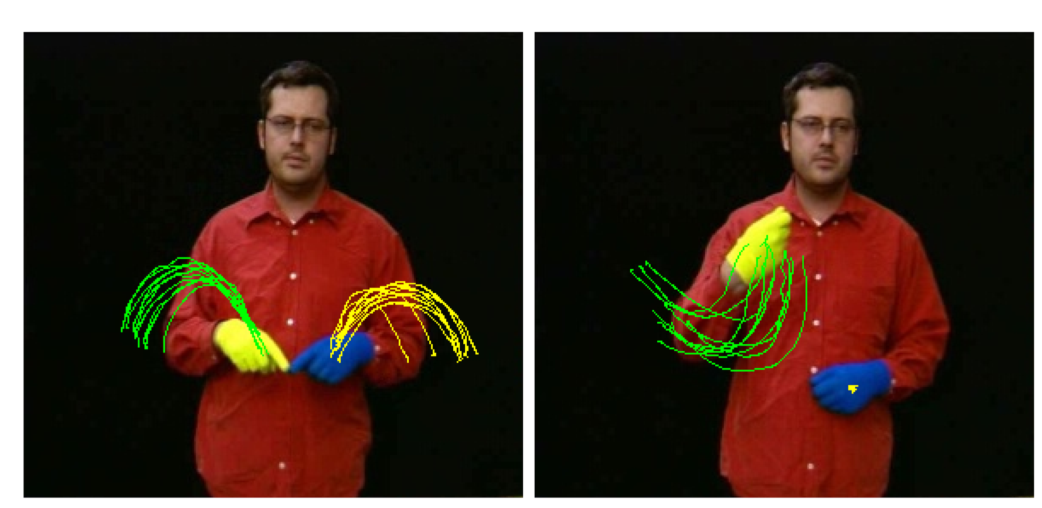
Recognition started 10 frames before the beginning of the gesture to examine the effect of co-articulation. The histogram shows the result for the experiment using only position and contour information in light color and in dark color the result when integrating position, contour and texture information.

Training examples



The trajectories and static hand gestures of the signs excited_interested (left) and live (right) are very similar. Therefore the shorter sign live dominates the recognition of the excited_interested performance. The integration non-manual observation like a grammar or facial expression should help to differentiate between similar signs.

Training examples



The signs "different" (left) and "bat" (right) shown with the trajectory differences of ten repetitions by the same signer.