American University of Sharjah
Department of Computer Science and Engineering

Learning the Gaze Direction using Bio-inspired Neural Networks

This Project is submitted to the Department of Computer Science and Engineering at the College of Engineering in partial fulfillment of the requirements for the degree of

Bachelor of Science Degree in

Computer Engineering

Presented By

Student 1: Diaa Emam, B00089017, COE

Student 2: Omar Odeh, B00082635, COE

Student 3: Ezekiel Morata, B00089094, CMP

Student 4: Mohammed Ali, B00088658, COE

Advisor: Dr. Aliaa Moualla

11/24/2024

Table of Contents

List of Tables

Table of Figures

Abstract

This project aims to improve human-robot interactions by developing a robotic system that can mimic human gaze behaviors. It is driven by the demand for robots with more social presence that aim to bridge the gap between existing robotic capabilities and the natural dynamics of human interaction. The approach combines a bio-inspired neural networks model with a robotic neck mechanism and other components, leveraging on neuroscience knowledge of head direction (HD) cells to effectively estimate and imitate human gaze directions. The advantages of this method include more intuitive interactions between humans and robots, increased engagement in social contexts, and improved facilitation of nonverbal communication. This substantially enhances robots' ability to perceive and respond to nonverbal cues in their surroundings.

1. Introduction

The presence of robotic applications is growing in the current market. There are numerous Industrial applications in which high-tech robots are used to perform complex tasks such as recycling and segregating different materials of garbage. However, none of these applications can effectively mimic human eye gaze as there are limited resources that enable robots to learn gaze direction. The ability of the robot to look at people naturally enhances the robot's social presence and fortifies its relationship with humans. That is what inspired the team to construct a bio-inspired neural networks model that utilizes a robotic head to assist in performing such a complex task. By studying how individuals observe, analyze, and respond to visual inputs through eye and head movements, the team aims to build a robotic system able to identify and respond appropriately to human attention cues. Additionally, understanding where a person is looking serves as a cornerstone of nonverbal communication and a critical enabler of joint attention, which is essential for human development and learning. Nonverbal cues like gaze direction convey rich contextual information, such as focus, intent, and emotional state, creating a foundation for social interactions. This capability is pivotal in fostering joint attention—a shared focus on an object or event between individuals—which is a developmental milestone and a precursor for learning complex behaviors such as language acquisition.

Research shows that infants and children rely heavily on gaze-following behaviors to infer meaning, build vocabulary, and understand social contexts. By enabling robots to mimic and respond to gaze cues, the project advances the integration of robots into human environments, allowing them to participate in these nuanced social exchanges. Such systems not only promote seamless human-robot collaboration but also pave the way for robotics applications in education, caregiving, and therapy, where joint attention is a critical element for engagement and learning. This would promote a more friendly interaction between robots and humans to normalize robot integration in a social environment, which is a critical objective of the project.

1.1 Motivation

As mentioned earlier, current high-developed robotic systems are used in Industrial tasks. However, most of those robotic systems were not intended for social purposes, but rather for work settings. The current social robotic systems were designed to perform simple predetermined tasks (i.e. waving to users), and some perform complex tasks (i.e. reacting to human emotions). Nonetheless, the encounters with these robots may appear artificial and fragmented, impeding the advancement of truly natural human-robot communication. Hence, due to the lack of advanced social robotic system features, specifically human gaze mimicking, the team was inspired to work with robots and develop a model that can effectively mimic human gaze direction using bio-inspired approaches.

There are many different methods and technologies for robotic gaze predictions. Nonetheless, they frequently lack the complexity needed to produce gaze behavior that is similar to that of a human. Moreover, these methods can offer basic gaze estimates but they might not be able to adjust to changing conditions or take into consideration the intricate interaction between head and eye motions. To achieve a more realistic and adaptive gaze behavior in robots,the team will try to integrate bio-inspired neural networks which imitate the neurological mechanisms behind human gaze control.

There is still a disconnect between the sophisticated gaze behavior displayed by humans and the capabilities of existing robotic systems, even with the progress made in gaze prediction technologies. To help close this gap and assist robots to effectively mimic human gaze behavior, the team opted to create new methods that make use of bio-inspired neural networks and include insights from computer vision and neuroscience. Our study is to help facilitate a more natural and efficient human-robot communication, which will have ramifications for a variety of social robotic uses, particularly human gaze imitation.

1.2 Project Description

Gaze direction is crucial in guiding attention, in both humans and possibly robots. By positioning cameras, robots can mimic the attentional process seen in humans. This involves directing their visual input mechanisms towards particular objects or areas of interest, demonstrating where their attention is focused. Even though robots may not have awareness or personal feelings, they can use eye direction as a useful way to show attention and aid in successful communication. This task would only be possible after achieving a few milestones. The team must first acquire an accurate dataset of the human gaze movement of different subjects. Then, create a bio-inspired neural network model for gaze detection and mimicking. Finally the team will train the model to understand these gazes and redirect or adjust the robotic head accordingly by correlating the head and eye movements. This would help in achieving our goal of improving human-robot interaction and promoting more natural communication between humans and robots by creating a complex gaze imitation model.

The main objective of the project is to construct a neural network model on a robot to be able to mimic and reciprocate a suitable response to the human gaze. Our project contributes to the development of a more developed robotic system for social interaction. By effectively analyzing and returning human gaze cues, the robot becomes able to respond to human social cues. This may help increase the sociability between robots and humans, resulting into a more interesting and meaningful interactions.

1.3 Organization of the Report

This report is divided into various sections. **Section 1 introduces** and explains the reasons for the project and outlines the **project's goals** and **objectives**. **Section 2** discusses and explains the **problem statement** and **design objectives**. It also addresses the issue the project seeks to resolve, suggests a resolution, and details the particular goals directing the project's development process. In addition, the team takes into consideration any **restrictions** or boundaries that impact the project's goals. Furthermore, **Section 3** provides a summary of current research, theories, and findings related to the project in the **literature review**. The current state of knowledge in the field are highlighted to offer context and backing for the proposed solution. **Section 4** covers the details of **system specifications**. The system's capabilities and quality attributes are specified by outlining both **functional** and **non-functional** requirements. **Section 5** discusses the **technical approach** and **design choices**. In addition, it outlines the technical plans for the proposed solution, such as block diagrams, workflow diagrams, and other design alternatives. **Section 6** includes the **project management plan** detailing the project's schedule, key dates, resources, and roles to guarantee successful project completion. **Section 7** details the **implementation** phase, where we describe the **hardware** and **software** components, as well as how they are integrated into the system. **Section 8** describes the **testing plan and expected results**. It outlines how we plan to assess the system's performance through offline and real-time experiments, detailing the performance metrics we will use to measure the system's success. **Section 9** covers the discussion of the **validation**, **verification**, and **performance analysis plan**. Guidelines for assessing the system's functionality and performance are established. **Section 10** analyzes the possible **global**, **economic**, and **societal impacts** of the project, assessing its broader implications outside of the technical aspects. **Section 11** covers the **standards** followed throughout the project, including ethical, hardware, and software standards. Finally, **Section 12** concludes and outlines the main discoveries, successes, and significance of the project.

2. Problem Statement and Design Objectives

2.1 Problem Statement and Proposed Solution

The model for the project was the study of neuronal activity in the superior colliculus of the cat by Peck [1], which shows the complex connection between head and eye movements in biological systems. Despite recent advancements, robots are still unable to mimic human gaze direction fully. Their gaze lacks characteristics that indicate purpose or attention, even though they can track objects. To better human-robot interaction, more research in Artificial Intelligence (AI) and brain science is required to replicate human-like gaze. Thus, the bio-inspired features of Peck's findings highlight the importance of understanding brain mechanisms to improve gaze estimation algorithms.

The main goal of this project is to create a neural network model that uses a robot head as an observation means to accurately predict the direction of the human gaze, inspired by biological mechanisms. Moreover, the team aims to improve the performance and applicability of gaze estimation algorithms in human-robot interaction scenarios by leveraging concepts found in biological systems and robotic components. The team will prioritize the incorporation of eye movements into the neural network model as a component of the project. By including this element of human eye movements, the team strives to improve the precision and authenticity of the robot's communication with people. This fusion will entail examining eye movement patterns and their relationship with head movements, combining knowledge from biological studies and AI methods to develop a more thorough model of gaze behavior similar to humans.

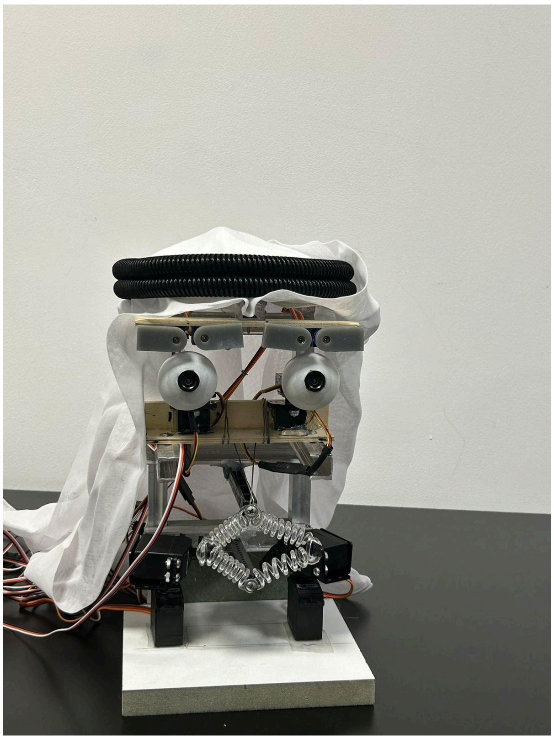


Fig. 1. Robotic Head

Our proposed system includes 3 main components: A platform, Dynamixel actuators, and a robotic head. The platform will be augmented with the Dynamixel actuators along with the robotic head, which will represent the robot's body. Furthermore, the Dynamixel actuators are used to improve the robot's ability to move and handle tasks, providing accurate and flexible control of its movements. Furthermore, the robotic head precise positioning and movement of sensors or cameras, mimicking the natural range of motion of a human head. The camera the team will be utilizing is the Mini Wide Angle FPV Camera as it provides high resolution (1080p), a Field of View (FOV) ranging from 120-170 degrees, and only weighs about 10 grams, making it possible for the robotic head and actuators to handle. This camera will be attached to the robotic neck to be able to capture live video footage of the user's head and eye movements. By combining these components, the team will create a robotic system that exhibits a bio-inspired behavior and interact with its environment.

Fig. 2. Mini Wide Angle FPV Camera

Studying robustness and adaptability in real-world conditions is crucial. Therefore, the team will evaluate the robustness of the model to factors such as changes in lighting conditions, background noise, and variations in head posture. This evaluation ensures the model remains effective and reliable in different environments and scenarios.

2.2 Design Objectives

The team will try to improve the capabilities of the neural network models involved with gaze direction to have more sophisticated human-robot interactions. In addition, reaching greater accuracy in perceiving human gaze and allowing for improvements in the implementation of developing intelligent systems is a challenge the team will face and try to overcome.

Perceiving human insight is an objective of the project. The social robot shall be able to function under various environmental situations, such as a cold room, or outside in the heat without being covered in shade. Furthermore, the team aims to build a model with a similar reaction time to that of humans, to mimic the average head-to-head movement speed of humans. The team believes that the social robot will provide advancements to current human-robot interactions. Both the hardware and software components of the project shall remain in good condition for further research. Finally, the neural network implemented into the robot shall be acceptable and user-friendly.

2.3 Limitations

The first possible limitation is the complexity of the human gaze. Due to the complex nature of the human gaze, it may pose a challenge in effectively simulating the model, even with focused attempts to enhance algorithms. In addition, it could be challenging to fully recreate elements like subtle eye movements and individual variations in gaze behavior. As there’s still more research to be done to have a complete understanding of the nature of the human gaze, the model would only be able to mimic the human gaze to a certain accuracy with the available data and research.

Moreover, individual body shape and clothing color may alter the model's ability to detect and interpret human gaze. Variations in body proportions may affect line-of-sight estimation methods, darkly colored or patterned clothing may generate visual noise, potentially confusing the model's tracking abilities. These variables emphasize the need of training the model on a diversified dataset that includes a variety of body forms and clothing styles in order to ensure reliable performance in a variety of real-world situations.

Another limitation could be the environmental factors affecting the accuracy of the model. The project's capacity to precisely recognize and duplicate human gaze may be affected by outside factors such as changes in lighting, objects in the field of view, and background images that might confuse the model. Thus, these environmental restrictions might make it difficult for the model to function effectively in real-life situations.

Furthermore, the project may face challenges in managing the variability and unpredictable nature of human behavior, despite efforts to improve the accuracy of detecting human gaze. Variations in individual preferences, situational environments, and cultural differences can all affect how well the project interprets and reacts to human gaze inputs.

Finally, the availability of exact motors may limit the project's ability to imitate the human gaze precisely. These motors, which are essential to precise movements, are expensive and beyond the scope of our project. It's crucial to remember that completely simulating the complexity of the human gaze remains a difficult task, even with the finest and most precise motors currently on the market.

3. Literature Review

In this section, each team member reviewed scholarly works related to the project and compiled resources useful to develop a social robot capable of mimicking human gaze behavior by integrating bio-inspired neural networks to interpret eye and neck movements. This interdisciplinary endeavor draws upon insights from neuroscience, AI, and robotics to replicate the complex mechanisms underlying human gaze behavior in robotic systems.

3.1 Evolving Hebbian Learning Rules in Voxel-Based Soft Robots [2]

In this paper by Ferigo et al. (2022) delves into the integration of Hebbian learning rules into voxel-based soft robots, robots crafted from flexible materials such as silicone or rubber, that can change shape and adjust to their surroundings. These robots are created, built, and operated with a three-dimensional grid made up of pixels. Although voxel-based soft robots offer flexibility and adaptability, the inherent flexibility of soft materials may lead to variability and imprecision, posing difficulties in achieving consistent results. Furthermore, soft robots may experience durability problems as their materials may deteriorate over time or be more prone to damage compared to conventional robots. Additionally, soft robots may lack the required strength or rigidity for tasks involving heavy lifting. Therefore, the team decided to work with a conventional robot mounted with a camera, which would be better suited for a more precise and managed human gaze direction mimicking. Nonetheless, the key concepts discussed in this paper will help build our model as they align closely with bio-inspired neural networks and learning mechanisms observed in biological organisms. Below are some benefits of the Hebbian Rule on gaze prediction:

a) System Adaptability and Visual Stimuli Learning Over Time

Hebbian learning mimics the synaptic plasticity observed in biological neural networks, wherein synaptic connections strengthen with correlated activity [2]. This understanding provides a foundational framework for developing bio-inspired neural networks for gaze prediction, enabling the system to adapt and learn from visual stimuli and neck movements over time.

b) Associative Learning

Hebbian learning facilitates associative learning, allowing the neural network to establish connections between visual inputs and corresponding gaze directions [2]. By incorporating Hebbian learning principles, the gaze prediction system can autonomously learn and refine its predictions based on the observed correlation between visual stimuli and subsequent eye movements.

c) Robustness and Accuracy of the Model

The integration of Hebbian learning rules enhances the robustness and accuracy of gaze prediction models. By evolving Hebbian learning rules within the neural network, the system can effectively adapt to variations in environmental conditions, head movements, and gaze behaviors, resulting in more reliable predictions [2].

3.2 General Theory of Remote Gaze Estimation Using the Pupil Center and Corneal Reflections [3]

Understanding the intricacies of gaze estimation methods discussed in this paper is crucial for enhancing the accuracy and robustness of gaze prediction systems [3]. The following key aspects from the paper offer actionable insights for gaze prediction:

a) Pupil Center Detection

The paper explores various techniques for identifying the pupil center, such as image processing algorithms and machine learning methods. For example, algorithms based on thresholding and edge detection can be used to identify pupil boundaries in digital images or video frames. Additionally, machine learning algorithms, including convolutional neural networks (CNNs), can be trained to recognize the pupil center based on facial image datasets.

b) Corneal Reflection Analysis

Guestrin and Eizenman elaborate on methods to analyze corneal reflections to estimate gaze direction accurately [3]. One approach involves capturing multiple reflections from known light sources and triangulating the gaze direction using geometric principles. Another method discussed in the paper utilizes mathematical models to map the relationship between the observed corneal reflections and the corresponding gaze angles.

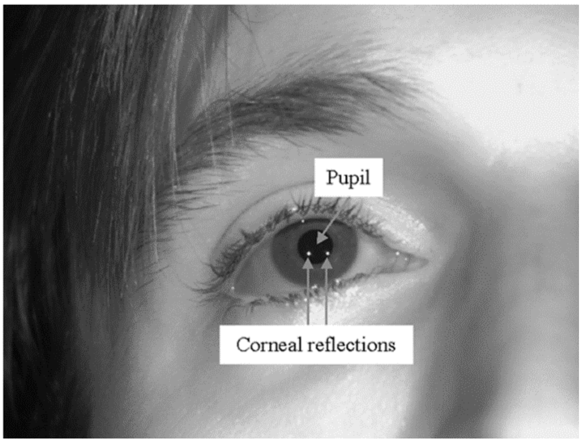


Fig. 3. Sample eye image showing the pupil and the two corneal reflections

Incorporating these methods into observational prediction models has the following benefits:

I. Improved accuracy

Implementing powerful changes to pupil center detection and skin reflection analysis improves the system's estimation accuracy, allowing it to predict the direction of the robot more accurately.

II. Non-destructive eye tracking:

The use of remote eye estimation techniques based on pupil axis and cone reflection enables non-destructive tracking of the robot's gaze, enabling natural interaction without the need for an intrusive sensor or device.

III. Real-world applications:

The conceptual framework presented in this paper provides practical guidance for implementing real-world estimation techniques, such as human-robot interaction, virtual reality systems, and assistive technologies [3].

3.3 A Novel Non-Intrusive Eye Gaze Estimation Using Cross-Ratio Under Large Head Motion [4]

Yoo and Chung (2004) present a novel approach for non-intrusive eye gaze estimation under large head motion using a method called cross-ratio. This overcomes the challenges associated with traditional gaze estimation methods, particularly in scenarios involving significant head movements. Even though the team will not be considering dynamic head movements in the model, understanding the methodologies proposed in this paper is essential for the project to be able to accurately predict eye gaze behavior under dynamic head motion conditions.

a) Cross-Ratio Technique

The cross-ratio methodology is presented by Yoo and Chung [4] as a novel method for non-intrusive eye gaze measurement. This method estimates gaze direction based on features of the eye, such as the center of the pupil and the corners of the eye, using geometric concepts. Regardless of the amount of head movement, the direction of glance can be precisely ascertained by computing the ratio of these locations. The problem with this method is the fact that five points or variables have to be calculated each time, which is very complex, inefficient, and time-consuming. That is why our proposed model will be more efficient and convenient as fewer variables will be of importance (i.e. head position).

I. Conceptual Foundation

The ratio of the lengths of four collinear points on a line segment is quantified by the geometric notion known as the cross-ratio. Cross-ratio can be used to locate important eye landmarks, such as the pupil's center and corners, in the context of eye gaze evaluation.

II. Landmark Detection

The first step in using the cross-ratio technique is to precisely identify certain eye landmarks in pictures or video frames. Usually, the middle of the pupil and the inner and outer corners of the eye serve as these landmarks. Robust landmark recognition can be achieved with a variety of computer vision approaches, including deep learning models or image processing algorithms.

III. Cross-Ratio Calculation

Once the eye landmarks are detected, the cross-ratio is calculated based on the relative positions of these points along the horizontal and vertical axes. Specifically, the cross-ratio is computed using the distances between the landmarks along these axes.

IV. Gaze Estimation

The direction of the user's sight is subsequently deduced using the computed cross-ratio value. The system calculates the corresponding gaze angle with respect to an axis or reference point by examining the variations in the cross-ratio as the user's gaze moves. Through calibration and training methods, known gaze directions are linked to certain cross-ratio values, establishing the relationship between the cross-ratio and gaze direction.

b) Robustness to Head Motion

One of the primary advantages of the proposed method is its robustness to large head motions [4]. Traditional gaze estimation techniques often struggle to maintain accuracy when the user's head moves significantly. However, the cross-ratio technique introduced by Yoo and Chung demonstrates promising results in maintaining accurate gaze estimation under dynamic head motion conditions.

c) Validation and Calibration

The effectiveness of the cross-ratio technique for gaze estimation is validated through experimental studies involving human participants. Calibration procedures are conducted to establish the mapping between cross-ratio values and gaze directions, ensuring accurate and reliable gaze estimation across different individuals and environmental conditions. Therefore, by understanding and implementing this technique, researchers can develop a robust gaze prediction model to further improve the human-robot interaction.

3.4 Gaze and Eye Tracking: Techniques and Applications in ADAS [5]

This paper investigates the use of eye tracking and gaze detection technologies to enable precise gaze mimicking in human-computer interaction systems. The study looks at the use of these technologies to understand human visual attention and gaze behavior better. Three significant resources are evaluated for their contributions to this topic, with a focus on giving insights and approaches for precise gaze imitation. Researchers want to develop systems that can accurately reproduce human gaze behavior in virtual worlds and human-computer interaction situations utilizing data collected through eye tracking and gaze detection techniques.

Eye tracking and gaze detection technologies have developed as important tools for studying human visual attention and behavior in a variety of domains. These technologies allow researchers to study where people gaze, providing significant information on cognitive processes, user interactions, and task performance. Accurate gaze estimate is critical in applications such as human-computer interaction, virtual reality, and augmented reality, where simulating human gaze behavior improves user experience and system efficiency.

I. The importance of eye tracking and gaze detection

The importance of eye tracking and gaze detection is underscored by its pivotal role in various fields, as highlighted in the previously examined study. This comprehensive overview delves into the intricacies of eye-tracking techniques and their applications within gaze-tracking systems. Central to this exploration is an understanding of the human eye's structure, including the differentiation between the visual and optical axes, pivotal for accurate gaze estimation as suggested in resource 4.

Incorporating data from eye movements such as fixations and saccades, researchers glean insights into visual attention, neurological conditions, and cognitive workload. Through meticulous analysis, eye tracking emerges as a powerful tool for discerning the point of regard (PoR) or gaze direction, enabling real-time monitoring in diverse applications. From medical psychology to human vision research and drowsiness detection for drivers, the breadth of eye-tracking applications is vast and impactful.

By providing precise gaze data, eye tracking facilitates advancements in understanding human behavior and enhances interactions in human-computer interfaces. As such, the first resource emphasizes the indispensable nature of eye tracking in modern research and technological innovation.

3.5 Evaluation and Integration of Gaze-Tracking Techniques for Accurate Mimicking [6]

The journal article evaluates different gaze-tracking techniques, focusing on feature-based and interpolation-based methods. It compares the performance of two web-based eye-tracking APIs, WebGazer and GazeCloudAPI, in terms of accuracy, reliability, and computational efficiency. GazeCloudAPI emerges as a more reliable solution, offering stable performance across different conditions and lower CPU usage compared to WebGazer. The study highlights the importance of choosing the appropriate gaze-tracking technique based on the application requirements and user preferences.

The article emphasizes the integration of eye tracking and gaze detection for accurate gaze mimicking in web applications. It provides a step-by-step tutorial for integrating GazeCloudAPI into web applications, emphasizing its ease of use and rapid prototyping capabilities. By leveraging gaze data obtained from GazeCloudAPI, developers can accurately replicate human gaze behavior in virtual environments and enhance the user experience. The resource underscores the potential of webcam-based eye tracking to democratize gaze-tracking technology and enable exciting applications in human-computer interaction.

3.6 A Survey of Robotics Control Based on Learning-Inspired Spiking Neural Networks [7]

In their thorough analysis of the development of robotics control using Spiking Neural Networks (SNNs), which are artificial neural networks that mimic the behavior of biological neural networks by processing input in discrete occurrences or "spikes", Bing et al. provide insights into the incorporation of biological intelligence concepts into robotics. They highlight how biological intelligence—which can be observed by the processing of information through spikes or impulses—significantly surpasses cutting-edge robots in a variety of real-world interaction scenarios[7]. Moreover, the study fills the gap by describing recent developments in computer science, electronics, and neuroscience that enable the development of hardware and software that will make it easier to create biologically realistic robots that are controlled by SNNs [7].

In addition, robotic applications are thoroughly classified and explained by the survey using various SNN learning rules, with a focus on how these rules affect the processing power, speed, and energy efficiency of robotic activities [7]. Additionally, it clarifies the platforms that are now in place to enable communication between SNNs and robotics simulations, advancing research and development in this area [7]. Moreover, by capturing the main reasons for using SNN-based controls in robotics and the promising prospects of this multidisciplinary field of study, this comprehensive forecast and review of previous advancements make a substantial contribution to the literature [7]. In conclusion, the paper offers a critical overview and projects future research directions and problems in the fast-developing field of SNN-controlled robot control [7]. This indicates the urgent need for comprehensive examinations.

3.7 A Rare Neural Correlations Implement Robotic Conditioning with Delayed Rewards and Disturbances [8]

In this paper, Soltoggio et al. research the brain principles underlying conditioning in robotics, addressing the challenges given by delayed rewards and the presence of distractions in dynamic situations [8]. It highlights the difficulties of establishing causality between actions and subsequent rewards, a problem that both biological neural networks and computational models face, emphasizing a theoretical gap in our knowledge of robotic neural conditioning [8]. The study proposes a unique technique that makes use of unusual brain connections to efficiently link actions or cues to subsequent rewards. This technique allows for sparse synaptic updates, which makes it easier to correctly identify routes that lead to rewards despite delays, as proved by human-robot interaction trials [8].

Moreover, the neural network shows both classical and operant conditioning features, indicating similarities in human and animal learning processes. The work done emphasizes the importance of synaptic tagging or eligibility traces in bridging the temporal gap between actions and rewards, with neuromodulation being crucial in reward-driven learning [8]. By utilizing these mechanisms, the study proposes a strong model for conditioning in robots that can navigate the difficulties of real-world interactions and variable timing [8]. This study contributes significantly to both computational modeling and our understanding of biological learning processes by providing a thorough review of neural mechanisms in conditioning and focusing on challenges and solutions for associating actions with delayed rewards in dynamic environments [8].

3.8 Gaze Redirection with Gaze Hardness-aware Transformation [9]

The article discusses the features of a model that allows for repositioning of the gaze's direction, named fine gaze redirection. "Gaze redirection is a computer vision task that redirects the gaze direction of the face image toward the target gaze direction [9]", and the method proposed, by the paper, to overcome the limitations of implementing higher gaze direction recognition, Gaze Hardness-aware Learning (GHT); the new addition feature to enhance current static gaze learning algorithms [9].

GHT takes into consideration that the plane image is not latent, but more dimensional; GHT creates views that cannot be expressed just with the rotation process and the gaze hardness-aware transformation from initial position to new direction alone [9]. The import of linear interpolation to generate a new formula for finding gaze representation after necessary adjustments to the destination of the predicted gaze. The suggested formulas provide generated gaze features, as well as removing other irrelevant gaze features such as head-pose. Another feature added was metric loss to minimize the effectiveness of head-pose and other unimportant features, allowing for finer gaze redirection data; the SG Loss function [9]. This function evaluates data to be if the predictions are off from actual testing, then the loss function is high.

3.9 Coordination of the eyes and head during visual orienting [10]

Precise coordination between the eyes and head is necessary for visual orienting in order to change the line of sight. When the head is stationary, reorientation happens through quick eye movements known as saccades. Under head-restrained conditions, there is a clear understanding of these movements with established patterns in saccade amplitude, duration, and peak velocity. Nonetheless, with the head able to move independently, the coordination between eye and head movements changes, leading to the need for modifications in current models.

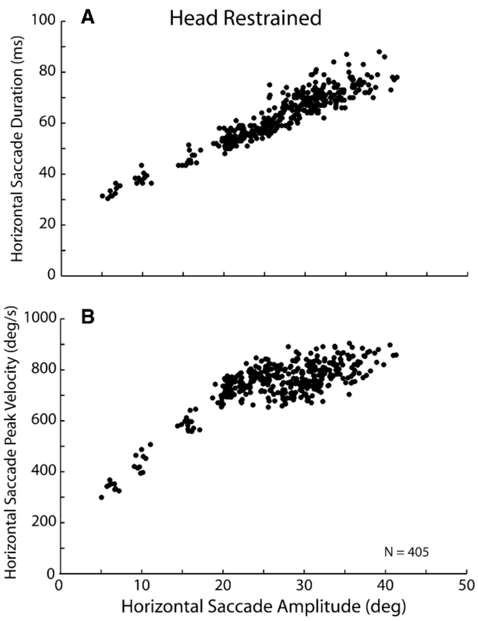


Fig. 4. Saccadic eye movements under head- restrained conditions

The figure above shows the typical traits of saccadic eye movements when the head is restrained, including the direct correlations between saccade distance and peak speed or length of time. Panel A demonstrates that when saccades are made horizontally within the range of 5 to 45°, the duration of the saccade increases proportionally with the increase in movement amplitude.

a. Eye-Head Coordination in Gaze Shifts

When the head and eyes move together to redirect gaze, their coordination alters the usual features of saccadic eye movements. For example, head movements play a more important role in larger gaze shifts by decreasing saccade velocity and extending movement duration. As the distance of gaze shift grows, the involvement of the head also grows, however, eye movements stay consistent.

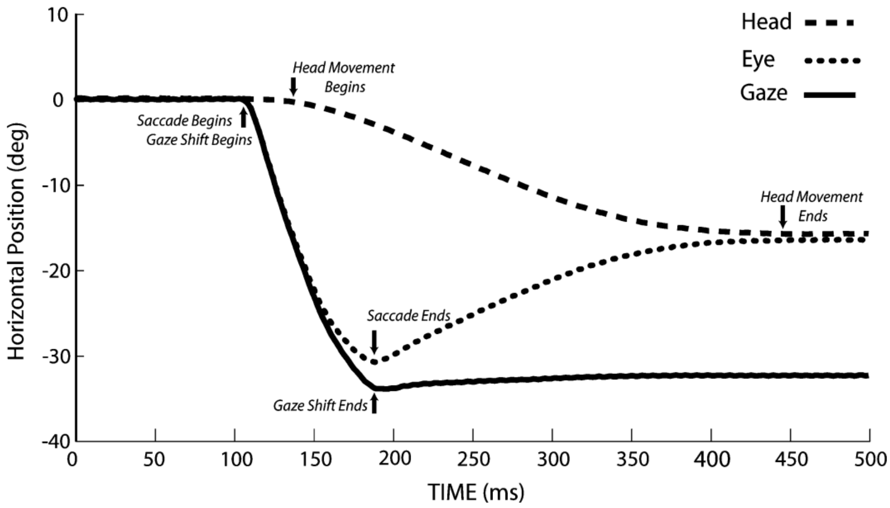


Fig. 5. Trial example of a gaze shift under head-free conditions

The above figure illustrates a sample of one attempt to shift one's gaze, including the eye and head movements, with the head unrestricted in its movement. Gaze shows the line of sight in relation to an external reference, Head indicates the orientation of the head in relation to an external reference, and Eye denotes the position of the eye in relation to the head. The combination of Eye position and Head position at every time sample results in Gaze position.

b. Modifications to Gaze Control Models

Recent models suggest that the coordination between the eyes and head in visual orienting entails interactions that are more intricate than previously believed. For example, alterations in head speed impact the speed of saccadic eye movements, indicating a connection between the motor signals for both, although they are not completely reliant on each other. This interaction clarifies why there is a range of eye and head movement patterns when non-human primates shift their gaze, as seen in Freedman's study from 2001.

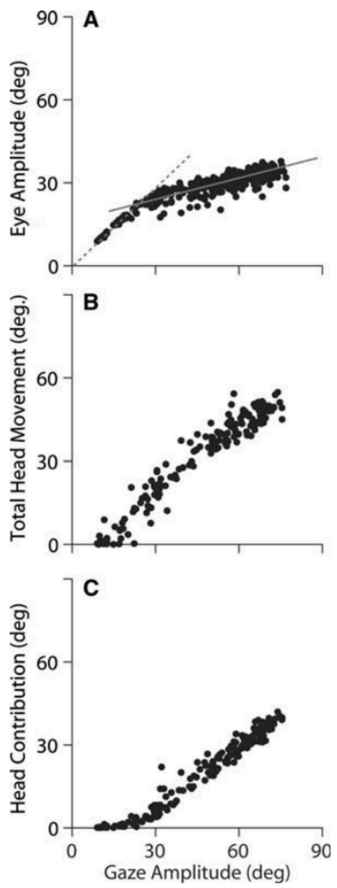


Fig. 6. Relative contributions of the eyes and head to gaze shifts of different amplitudes directed along the horizontal peak

The figure above visualizes how the head begins to contribute more during larger gaze shifts and how the contribution of the eyes begins to level for these movements. Panel A shows the eye movement amplitude (in degrees) as a function of gaze shift amplitude. Panel B plots the total head movement amplitude for the same gaze shifts. Panel C shows the head's contribution to the gaze shift (as opposed to total head movement) — the amount of head movement that directly aids in shifting the gaze. For small gaze shifts, the head plays a minimal role, but as gaze shifts exceed 20°, the head begins to contribute significantly.

c. Kinematic and Neural Mechanisms

Various models have been used to represent how the neural control mechanisms of coordinated eye-head movements work. Initial theories suggested that one single command for gaze velocity controls both the eyes and head simultaneously. Yet, newer models propose separate control signals for the eyes and head that work together to enhance gaze shifts. Moreover, the vestibulo-ocular reflex (VOR), which usually helps maintain vision stability when the head is moving, is adjusted during gaze shifts to avoid conflicting with the commands for eye movement. These models are shown below.

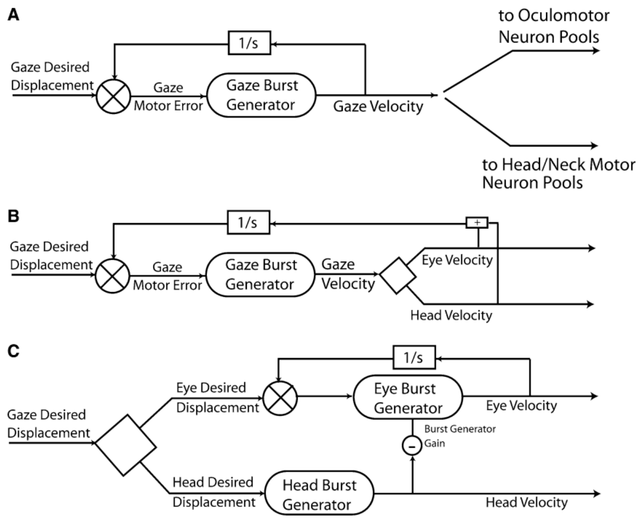


Fig. 7. Schematic diagrams of three classes of gaze control model

From the above schematic, it can be seen that in model A, the desired gaze movement is matched against an internal assessment of current gaze movement to create a signal for gaze motor error. One gaze burst generator creates a signal that is directly related to the speed of the gaze, and this signal controls the movements of both the eyes and the head. The model in panel B suggests the gaze velocity signal is divided into distinct eye and head velocity commands within the dynamic feedback control loop. Finally in model C, the breaking down of gaze signals happens before the control loop. This involves using distinct burst generator elements driven by desired eye and head displacement signals to generate commands for eye and head velocity. A signal that varies directly with the speed of head movements is suggested to decrease the amplification of the saccadic burst generator. This interaction between the head and eyes enables these hypotheses to account for the changes in saccade speed that happen as head movement speeds go up.

3.10 Robustness Study of a Multimodal Compass Inspired from HD-Cells and Dynamic Neural Fields [11]

This paper by Delarboulas et al. introduces a bio-inspired model to address robot orientation in navigation tasks. This model is based on the neurological function of head direction (HD) cells, which merge internal (idiothetic) and external (allothetic) sensory information for head orientation. The paper provides a plausible model that uses a multimodal compass that integrates sensor data to enhance the robustness and reliability of the vision-based navigation systems in a dynamic setting.

a) Multimodal Compass and Dynamic Neural Fields for Navigation

The model is made of a multimodal compass that mimics the neurobiological function of HD cells. Based on the PerAc architecture [12], this model employs dynamic neural fields (DNFs), which handle decision-making in real time. DNFs, introduced by Amari [13], integrate sensory inputs from multiple modalities and fuse them to produce a reliable navigation direction. In this model, the DNF integrates visual, magnetic, and proprioceptive data to compute a consistent head orientation. as shown in the block diagram below.

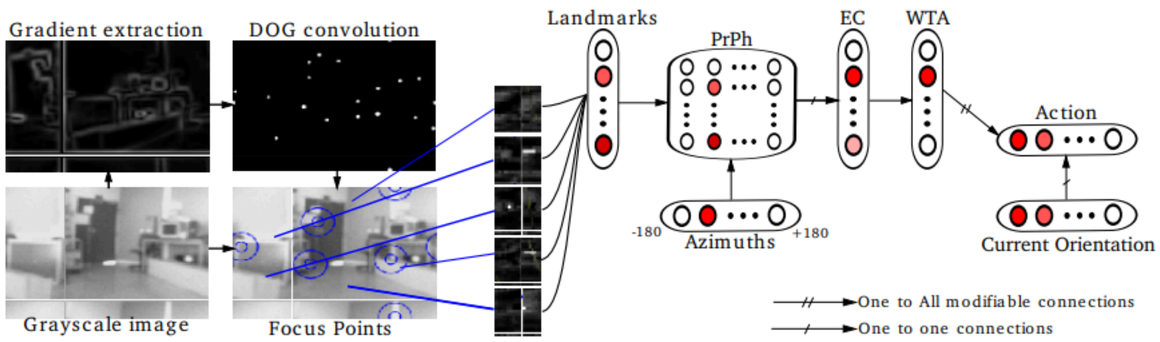


Fig. 8. Block diagram of the global navigation architecture for visual navigation

The model by Delarboulas et al. combines this architecture with their own neural network for the multimodal compass as shown in the above figure.. The figure illustrates the Visual Compass and Sensors Fusion modules that form the core of the multimodal navigation system. The Visual Compass receives an input from visual landmarks and computes a gaze direction θL based on visual inputs and previously learned landmarks. The Sensors Fusion module then integrates this visual data with allothetic (external cues) and idiothetic (internal cues like proprioception) inputs to form a more comprehensive estimate of the robot's head direction θH.

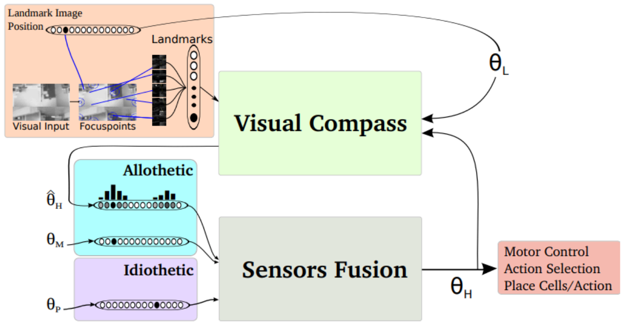


Fig. 9. Block diagram of the orientation system as proposed by Delarboulas et al.

Sensors fusion merge allothetic information with idiothetic to compute a reliable ˆθH. The shift between Allothetic and Idiothetic modalities is sent as input to a Dynamic Neural field driven by the amari equation. Properties of the DNF allow to keep a consistent measure of this shift. The visual compass is particularly useful in familiar environments where landmarks are available and the sensors fusion layer allows for robust navigation when visual cues are unavailable or unreliable. The researcher also employed Winner-Takes-All (WTA) strategy which would further ensure that only the most salient landmarks influence the robot's action.

b) Visual Navigation and Place-Action Associations

The visual compass relies on identifying and associating visual landmarks relative to the robot's location. Previous studies by Gaussier et al. [2], demonstrated the effectiveness of landmark-based navigation, Delarboulas et al. expand on this by integrating a short-term memory system into the visual compass, which stores landmark-based predictions of orientation and merges them with idiothetic inputs in the DNF.

Through place-action associations, the model links visual landmarks with specific actions. This simple learning mechanism allows the robot to perform homing or route-following tasks autonomously. The robustness of this system is illustrated in experiments where the robot maintains accurate orientation, even in the presence of environmental noise or perturbations like electromagnetic interference.

Overall, The bio-inspired model proposed by Delarboulas et al. provides key insights in the field of bio-inspired robotics by providing a resilient and adaptable navigation system.

3.11 Internally-Organized Mechanisms of the Head Direction Sense [14]

Peyrache et al. (2015) investigate the dynamics of head direction (HD) cells, which act as a neural compass by increasing firing rates in response to the animal's head orientation. By monitoring HD neuron activity in mice's antero-dorsal thalamic nucleus and postsubiculum, the study distinguishes between stimulus-driven and internally produced neuronal activity. The findings show that HD neurons maintain their temporal correlation structure even while sleeping, highlighting the impact of internally structured networks on the accuracy of directional signals.

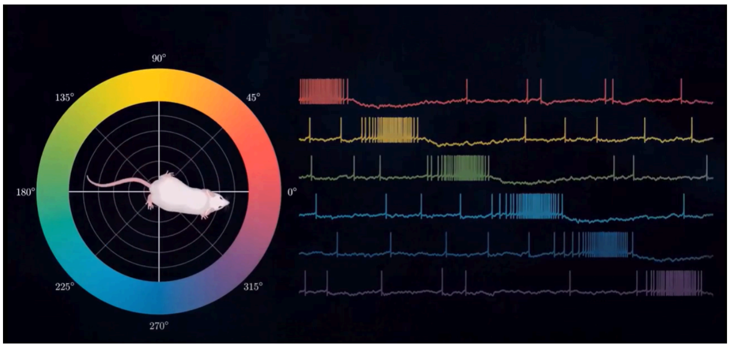


Fig. 10. Neuronal firing patterns correlating to different head angles of a mouse

The study's accompanying Figure 11 illustrates in detail the dynamics of head direction cells. It includes a polar plot with a mouse in the center, a color gradient showing various directional headings, and graphs on the right that show neuronal firing patterns correlating to different head angles. This figure demonstrates how neural activity changes with head direction, emphasizing the complex link between spatial orientation and neuronal firing. Such representations are critical for comprehending the complicated connections that occur inside brain circuits involved in spatial cognition.

3.12 Gaze Tracking by Joint Head and Eye Pose Estimation Under Free Head Movement [15]

The paper proposes a different approach to the gaze tracking algorithm as a function of both eye and head pose components, without requiring prolonged user cooperation before gaze estimation. It exploits the trajectories of salient feature trackers spread randomly over the face region to estimate the head rotation angles, which are then used to drive a spherical eye-in-head rotation model that compensates for the changes in eye region appearance under head rotation.

Frontal eye and head poses are initially detected from the relative distances between the face features, permitting the extraction of the frontal eye regions, f_{x0}, which x_0 denote the iris center coordinates. [15] Their method entails obtaining facial photos and estimating the 3D head pose, while also analyzing the ocular region to establish the eye position. These estimates are then merged to generate the gaze vector.

Fig. 11. Feature Tracking on the face: Weighted Medium (Red), positions above the eye (green), and positions below the eye (yellow)

The figure illustrates detecting the facial features before and after repositioning their head in real-time, as feature trackers latched for tracking. The head poses are estimated using the filtering process; Kalman and Particle, with particle filtering, contribute more than the Kalman filter. The rotation of the frontal reference image by the head pose angles [15], even accounting for the head translation for eye alignment. The gaze estimations are computed by a summation of the head and eyeball rotation estimations.

3.13 Conclusion

Overall, the examined articles offer perspectives on improving gaze estimation and interactions between humans and robots. Furthermore, understanding the Hebbian learning principles will help in enhancing the team's understanding of how neurons or cells strengthen or weaken over time. This will help in building the model and will improve its adaptability and resilience. Moreover, research on methods for estimating where someone is looking lays the groundwork for precise gaze forecasts, while progress in robot control through Spiking Neural Networks presents possibilities for creating robots that mimic biological systems. In addition, creative methods such as GHT, improve the accuracy of gaze estimation models by adjusting the gaze direction. Collectively, these contributions are laying the foundation for enhanced human-robot cooperation and interaction across a range of uses.

4. System Specification

4.1 Functional Requirements:

FR 1. Video Analysis and Gaze Identification:

FR 1.1. The system shall employ a state-of-the-art CNN model to analyze video data and identify the gaze coordinates of the eyes and head.

FR 1.2. The system will create a dataset with various head and eye positions captured from different angles for effective gaze estimation.

FR 2. Neural Network Investigation:

FR 2.1. The system must investigate Neural Networks (i.e. Spiking Neural Networks or SNNs) for analyzing visual data from cameras to detect subtle eye and head movements.

FR 2.2. The system will train the model with diverse head and eye positions for accurate gaze estimation.

FR 3. Eyes and Neck Coordination and Adjustment:

FR 3.1. The system shall track both old and new coordinates of human head and eye locations for detecting changes in gaze direction.

FR 3.2. The system will compute the difference in coordinates to adjust the camera to replicate human gaze movements accurately.

FR 4. Testing and Precision:

FR 4.1. Testing with unseen data shall be conducted to confirm precision and make architecture adjustments if needed.

FR 4.2. The system shall maintain precise timing for adjusting the robotic neck/camera orientation for smoother interactions between the robot and humans.

FR 5. Limitations and Reliability:

FR 5.1. The effectiveness of the gaze learning model shall be limited to the efficiency of the TurtleBot3 employed to mimic the human gaze direction to only a certain degree of freedom.

FR 5.2. The robot will operate reliably only when tasks are assigned and interact with users equipped with gaze technology.

FR 6. Maintainability and Responsiveness:

FR 6.1. Maintainability is essential to ensure efficient project functioning.

FR 6.2. Having different components for the robot will allow the system to respond quickly and accordingly to input data and gaze redirection positions.

4.2 Nonfunctional Requirements:

NFR 1. Minimize Cost and Hardware:

NFR 1.1. The system must utilize only a camera for data collection, reducing cost and eliminating the need for additional hardware components.

NFR 2. Human Interaction and Natural Behavior:

NFR 2.1. The system shall be designed to exhibit natural human gaze behavior, facilitating easy interaction through nonverbal cues.

NFR 2.2. Users will have hands-free control, with the ability to initiate or terminate interactions immediately.

NFR 2.3. The system must effectively mimic the human gaze to a certain degree such that the interaction between the robot and the human appears natural.

NFR 3. Performance:

NFR 3.1. Performance depends on the effective integration of multiple neural networks used for modeling the gaze of a human and mimicking the gaze direction.

NFR 3.2. The system must ensure consistent performance across various lighting conditions and user distances from the camera.

NFR 3.3. The robot must provide consistent gaze imitation under various predictable environments during interactions with human users.

NFR 3.4. The response time must be swift; the robot's directional gaze shall react within 500 milliseconds from the human gaze.

NFR 4. Safety and Reliability:

NFR 4.1. The robot will employ multiple fuses to prevent damage to critical components in case of unnecessary surges in the circuit current.

NFR 4.2. The speed of actuators and motors, as well as the degree of movement of the robot, shall be limited to prevent sudden or erratic movements that could harm its surroundings while still allowing natural gaze imitation.

NFR 5. User Interface and Integration:

NFR 5.1. A straightforward, user-friendly interface is a must to promote the robot's capabilities and facilitate control and monitoring of its actions and behavior.

NFR 5.2. The system should seamlessly integrate with hardware components like the camera, along with software modules like CNN and other neural networks, to ensure smooth operation.

NFR 5.3. The system must support reliable communication protocols to facilitate interaction between different components, ensuring data integrity and timely exchange of information.

NFR 5.4. The interface should account for different user-specific factors to make it easier for users to adjust neural network parameters and achieve the desired performance.

5. Technical Approach and Design Alternatives

This section will discuss the technical approaches the team will use along with possible solutions and their alternatives, and finally decide on an approach.

5.1 Design of the Solution

5.1.1 Block Diagram

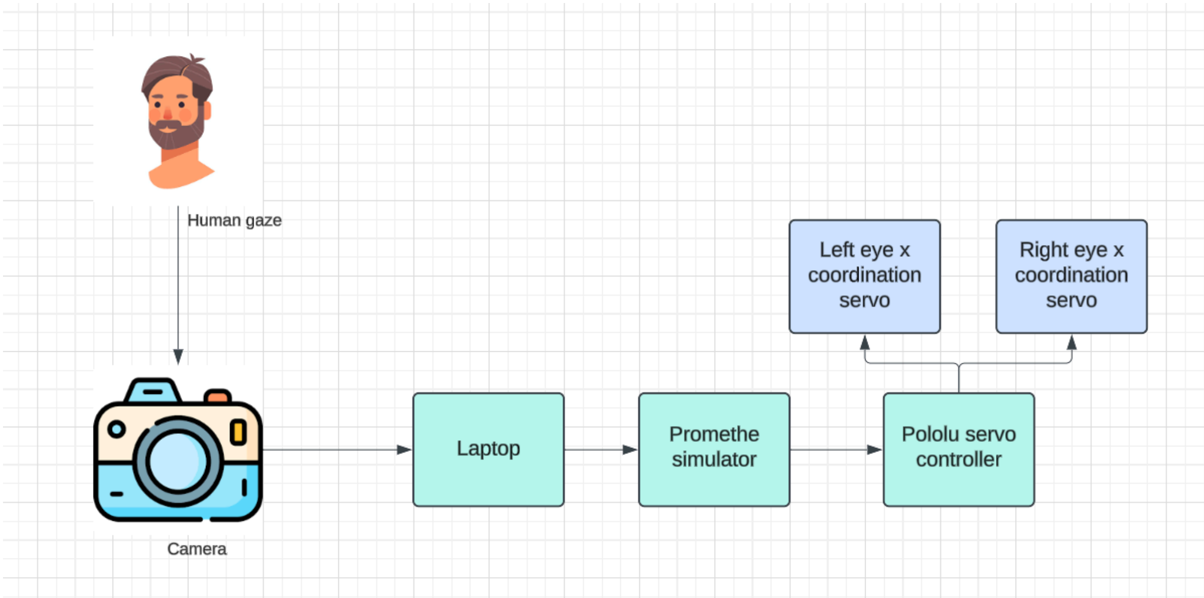


Fig. 12. Block diagram for Gaze mimicking robotic system

In the Gaze mimicking robot's system technical architecture, as shown above in Figure 12, there are six main components: The camera, linux based ubuntu computer, and two servos for the eye movement of the robot. The camera allows the robot to analyze the gaze of a human subject in order for the robot to mimic this gaze and move its face accordingly.

We chose a computer as it is able to run Linux natively as well as run Promethe simulation locally and also send instructions to servos through the Pololu mini maestro12 controller. The computer plays a pivotal role in the system as it analyzes data received from the camera and processes this information on the human subject's gaze direction. The information is then sent to the Bio-inspired neural network simulator Promethe for processing and simulating the results in real-time. Promethe then sends the necessary instruction to the servos through Pololu to move the robot's eyes accordingly, which is the reason why the system requires larger RAM and memory space provided by the Laptop.

The robot uses two servos to control the movement of the system. These Servos can be programmed to move instantaneously directly by the Pololu servo controller which acts as a translation layer for the servos to understand. Two of the servos control the horizontal of the eye.

5.1.2 Workflow Diagram

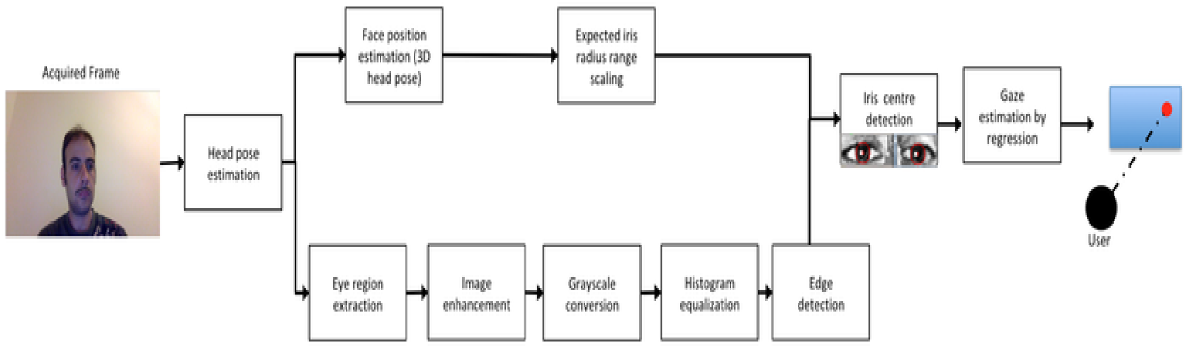


Fig. 13. Robot Gaze Estimation System Workflow Diagram [16]

The diagram above [16] illustrates the workflow for developing a robotic system capable of imitating the human gaze. Just as a workflow diagram provides a graphic overview of a business process, this diagram outlines the step-by-step procedure for creating the robotic system, from capturing an image with the robot's camera to accurately predicting the direction of the human gaze.

The first step involves inputting an image or frame into the system, which is captured by the robot's camera. Subsequently, the system calculates the head position, extracts the eyes, and analyzes the edges within the image to detect the iris center, a crucial step in accurately predicting the direction of the human's gaze.

To implement this system successfully, advanced tools such as neural network simulators are essential. An example of such a tool is Promethe, a widely recognized simulator frequently utilized in research, education, and the development of neural network algorithms and applications. By utilizing Promethe, the team can replicate and test the functionality of artificial neural networks, ensuring the accuracy and effectiveness of the robotic system in imitating the human gaze.

5.1.3 Flow-Charts and/or State Diagrams

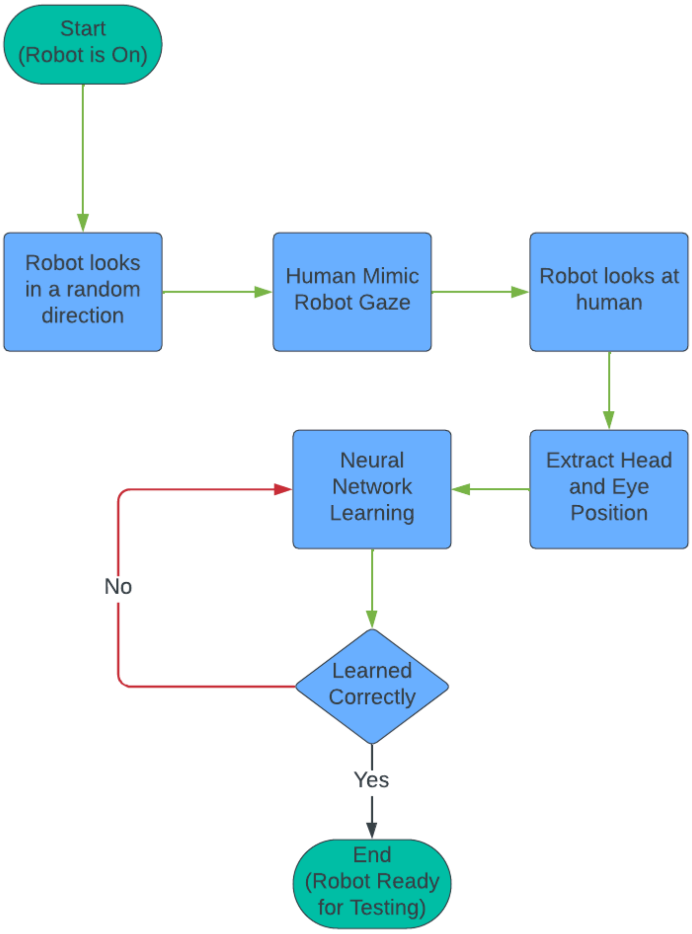


Fig. 14. Robot Training Flow Chart

The flow chart above shows how the robot would be taught to imitate the gaze actions of a human using the online learning system. At the outset, the robot lacks any prior knowledge. The training process begins with the robot glancing in a random direction. The user will then need to attempt to replicate the robot's eye movements. The robot will then observe the user, using the camera to determine the position of the human's head and eyes. This would lead to connections being formed between the individual and the imitative behavior. If the robot was trained correctly, it would successfully imitate the human gaze, otherwise, the team will continue enhancing the model until the desired level of accuracy is reached.

5.1.4 Main Resources Needed {on the block level}

Table 1. Main Components Required

Components	Description
Tilt Pan Movement	Mechanism for pivoting and rotating the camera to track the human gaze.
Servo Motors	Provide accurate angular control, which is required to move the tilt-pan mechanism and rotate the lens.
Mini Wide Angle FPV Camera	A device that captures visual information with 26x15x15 mm dimension and an approximately horizontal resolution of 1000TVL
Pololu mini maestro 12	A servo controller that takes in information from promethe and translates it to to instructions understandable by the servos
Simulator - Promethe	A neural network simulator is used to evaluate and enhance gaze estimation techniques.

Table 1 features all the main components that will be utilized for the bioinspired gaze learning project. The Pololu servo controller manages multiple tasks of relaying and adjusting the hardware components, servo motors, to the requirements of the system. Promethe simulator shall make use of neural network features implemented and enhance current gaze estimation techniques.

5.1.5 Use Case Diagram

Fig. 15. The Use Case Diagram of the System

5.2 Alternative Designs

When constructing a robotic gaze estimation system, it is critical to consider several techniques to enhance each subsystem's functionality. Based on our evaluation of the literature, the team is exploring various design alternatives to improve the system.

I. Tilt-Pan Mechanism Alternatives:

In reference to the context in [2], which outlines voxel-based soft robots and Hebbian learning principles, one could look into employing soft robotic components for the tilt-pan motion. However, because of the significance of accuracy in gaze tracking, the final design prioritized traditional mechanics over adaptable but less accurate soft robots.

II. Camera Alternatives:

3D Depth Cameras: according to [3] and [4], a 3D depth camera might provide additional data regarding object distance and pupil placement regardless of head movement, resulting in boosting gaze estimate accuracy.

III. Servo Motors Alternatives:

Stepper Motors: Stepper motors are a cost-effective and simple alternative to servo motors, suitable for applications where simplicity and price are preferred above high dynamic performance, particularly in educational and research environments [18].

IV. Actuator Alternatives:

Electroactive polymers (EAPs): EAPs are an effective substitute for standard actuators in robotic applications, allowing for considerable mechanical deformations in response to electrical inputs. These materials are ideal for soft robotics, as they provide flexible and sensitive actuation that mimics natural biological motions [19].

V. Promethe Simulator Alternatives:

Stuttgart Neural Network Simulator: RSNNS would be the alternative simulator instead of Promethe, allowing the use of features of neural network procedures [20]. The simulator runs with C++ and/or R language.

6. Project Management Plan

Name	Start Date	End Date
Research Phase	Aug 26, 2024	Nov 22, 2024
Weekly Meetings	Aug 26, 2024	Nov 25, 2024
Report Part I	Aug 26, 2024	Sep 09, 2024
Finalizing Lit. Review of Part I	Sep 09, 2024	Sep 16, 2024
Working on the System (SW/HW) Architecture…	Sep 09, 2024	Sep 30, 2024
Report Part II	Sep 09, 2024	Sep 30, 2024
System Integration	Sep 16, 2024	Oct 07, 2024
Prototyping Phase	Sep 30, 2024	Oct 21, 2024
Report Part III	Sep 30, 2024	Oct 21, 2024
Draft Final Report	Oct 28, 2024	Nov 11, 2024
Validating/Testing the Model	Oct 07, 2024	Nov 18, 2024
Peer Review & Final Report	Nov 11, 2024	Nov 18, 2024
Working on the Presentation	Nov 18, 2024	Nov 25, 2024

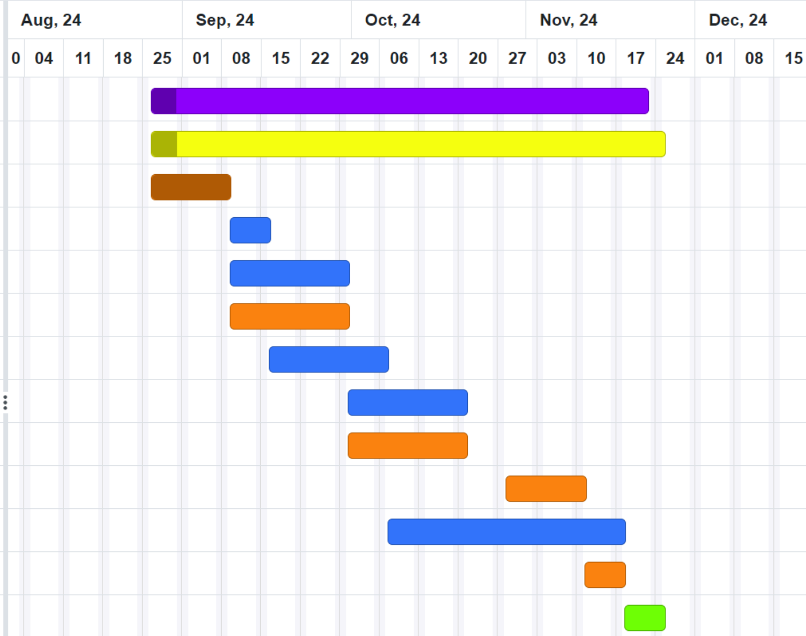


Fig. 16. Project Plan Gantt Chart

The above figure displays the project plan the team will try to respect throughout the year. In the first weeks of the project, the team will be mainly researching the latest discoveries on topics that benefit the project, such as gaze prediction/mimicking, robot neck, and head movement, and bio-inspired neural network papers. The researching phase of reading numerous scientific articles, as well as writing a literature review containing information researched by all members for gaze-direction technology and bioinspired network models that the gaze technology implemented upon. This phase will continue until the end of the semester.

The team has devised a weekly schedule with the advisor, updating us on researching topics related to the gaze direction technology and finding papers reinforcing the project. Topics include bio-inspired neural networks, hardware capable of eye and neck movements, and further research on gaze learning.

Moreover, the team will then begin to design the system's SW/HW architecture. In addition, the team aims to optimize cost-effectiveness in their efforts, as well as enhance the robot's interaction capabilities and its potential societal contributions, fostering realism within such engagements.

The final stage will cover the system integration, prototyping, and conducting comprehensive validation and testing procedures for the model. This meticulous approach aims to ensure the robustness and efficiency of the project solution. Rigorous testing will be employed to validate the model's functionality and adherence to predefined criteria.

7. Implementation

7.1 Hardware

1. Components Used

For our project, we’ll be using a variety of components to enable the robot to mimic human gaze direction. A **Linux-based machine** will serve as the main controller, managing and coordinating the various hardware components. To capture head and eye movements, we will use a **Mini Wide Angle FPV Camera** with a wide Field of View (120-170 degrees) and high resolution to ensure detailed video footage is available to support gaze direction imitation.

The robot’s eyes will be controlled by **four** Hitec HS-55 **servo motors** which will manage the tilt and pan motions of the eyes necessary to follow the gaze direction. These servos will be connected to a **Pololu Mini Maestro 12-channel usb servo controller** for smooth speed control and precise movement. Additional **actuators and servos** will be employed to handle the physical adjustments based on the neural network’s output, allowing for synchronized movement of the robot’s head and eye.

2. Servo motor connections with Pololu

As the robot being used in our research was initially developed by a previous group of seniors. Our current work builds upon their foundation, with a focus on advancing the robot’s capabilities in gaze imitation. The Maestro Control Center for Pololu software provides a user-friendly graphical interface that allows for straightforward configuration of each servo’s parameters such as range, speed, and acceleration. We will be connecting 5 servo motors, from channel 8 to 11. This setup will help us find all the center, maximum and minimum placements the servo motors can adjust to. We connect the servo motors to channel pins 8 to 11 on the Mini Maestro 12 controller to control the eye segments of the robot head, to identify which servo pins correspond to each eyebrow segment. Channel 8 corresponds to horizontal movement of the Left Eye, channel 9 corresponds to vertical movement of the Left Eye, while channel 10 corresponds to the horizontal movement of the Right Eye, and channel

11 corresponds to the vertical movement of the Right Eye. Once identified, we can connect each servo motor to its corresponding pin on the Mini Maestro 12. We will also connect Channel 7 to the servo that is connected to the bottom of the to handle the horizontal movement of the neck. Additionally, we would need to provide power and ground connections to the servo motors to ensure they receive the necessary power to operate, this achieved by connecting to the VSRV pins located on the Pololu board. Once the servos are connected, you can program the Mini Maestro to control each servo independently, allowing you to articulate the robot's eyes and neck movement according to your desired objective.

3. Camera Connections

Camera connections To set up the camera connections, we started by analyzing the wires coming out from the camera which were a 5V wire, a ground wire, a video signal wire, and finally an audio signal wire. For our design, we only need the 5V wire, the ground wire, and the video signal wire. The connection and the functioning of the camera should be through an AV wire to connect it to any functioning computer to share a live video from the camera. To start those connections, we began by connecting the 5V wire from the camera to the Arduino's power supply (5V) by welding with copper to ensure we are getting enough power supply to the camera. Moving on, we connected the ground wire from the camera to both the Arduino's ground. and to the AV wire ground cable, to connect the ground wire to the ground of the Arduino, we welded both wires together using copper to get a two-way connection for the ground wire, and we also welded the ground wire from the camera to the AV's wire ground, this allowed us to ensure a safe electrical connection to the wires. Furthermore, we welded the yellow wires coming from the camera to the AV wire itself, which will allow us to ensure a live streaming from the camera to the screen we are using when it is connected to the USB to AV converter.

Analysis of the performance of the implemented system

The performance of the implemented gaze-mimicking system is driven by the coordination between the linux machine and the Pololu servo controller, which serve as the primary control units, and the hardware components responsible for movement. The system relies on a camera to capture head and eye movements, which are processed locally by a linux machine to generate commands for the servo motors and actuators that control the robot's head and neck movements.

The servo motors, connected via an Pololu Servo Controller, are designed to respond to the commands issued by the machine, enabling precise eye movements in alignment with the captured gaze data.

In terms of responsiveness, the system performs consistently in stable environments. However, challenges arose in variable lighting conditions or when dealing with unexpected inputs, emphasizing the need to optimize the image processing algorithms for better adaptability.

7.2 Software

1. Entities

a. The Perceptron:

The perceptron (a linear classifier and single-layer neural network) is specifically designed for supervised learning. It mimics biological neurons by linearly combining input samples and applying an error correction method to adjust synaptic weights. The perceptron is particularly effective for linearly separable problems; however, it can struggle with more complex datasets. Because it can clearly distinguish between two data groups, it serves as a binary classifier, outputting either 0 or 1. Although the perceptron learning rule is foundational, it is given by the following formula:

$$\mathbf{W0i = Wi + \alpha(Yt - Y)Xi}$$

In this equation, wi0 represents the updated weight for the input, Yt denotes the target output. On the other hand, Y signifies the actual output; α refers to the learning rate, while Xi is the input linked to the weight. Additionally, Wi is the current weight [21]. However, understanding these variables is crucial because they interplay significantly in the learning process. Although this equation may seem straightforward, its implications are profound in the context of machine learning.

b. SAW (Selective Adaptive Winner):

The Selective Adaptive Window (SAW) is a variant of the K-means clustering algorithm, which is classified as unsupervised (and is utilized for data classification and grouping). Within a network of neurons, SAW initializes synaptic weights randomly; it subsequently identifies a winning neuron that embodies the first category. The "vigilance" parameter plays a crucial role in determining the level of detail within these categories. If the input changes and the activity of the winning neuron falls below the vigilance threshold, a new neuron is recruited (this process is essential for maintaining accurate categories). By adjusting the vigilance threshold through neuromodulation signals, the granularity of the learned categories can be modified during the operation of the SAW algorithm. A higher vigilance level results in stricter category definitions, however, a lower vigilance level provides a more lenient approach to categorization. Although this flexibility is beneficial, it requires careful management to ensure effective learning.

As learning advances, synaptic weights undergo adjustments over time. The equations governing these weight changes are as follows :

$$\Delta w_{ij} = \delta_j^k (a_j(t).E_i)$$

$$(k = ArgMax(C_j))$$

$$\delta_j^k = \begin{cases} 1 & \text{si } i = k \\ 0 & \text{sinon} \end{cases}$$

Fig. 17. Formulas for the Variation in Synaptic Weights Among Neurons [21]

However, they can be complex. Although the principles are straightforward, the actual calculations often present challenges, because they require a deep understanding of various factors. This intricate process necessitates careful consideration of each variable involved.

The third equation specifically modifies the weights of the winning category. When a neuron j is recruited, the function aj(t) becomes 1. Recruitment is random, preventing unlearning, as long as the neuron does not already represent a category [21]. During recruitment, the competition mechanism boosts the neuron's activity to 1, enabling only the synaptic weights connecting the recruited neuron to the input layer to be adjusted. These weights are modified to match the input pattern, allowing the neuron to "remember" the shape. The activity of the recruited neuron is influenced by newly acquired data. If the neuron recognizes a previously learned shape, it identifies it. However, if the shape is too different from what was learned, the neuron will not recognize it [21].

c. Hebb's Learning Rule (Delta Rule):

Hebb's learning rule (which draws inspiration from the mechanisms of biological neurons) reinforces synapses when the associated neurons fire together. This principle can be applied to the Perceptron, using the following weight update equation:

$$\Delta w_{ij} = \alpha . y_j . x_i$$

This equation is relevant and applicable when synapses are activated at the same time. However, the rule can become unstable, because coincidences might lead to uncontrolled increases in synaptic weights, resulting in unbounded growth. To counteract this, when neurons fire asynchronously, the synapse is either selectively weakened or removed; this prevents the accumulation of excessive weights. Normalizing weights (between 0 and 1) serves as another strategy to avert exponential growth, which can be achieved using an updated equation [21]:

$$\Delta w_{ij} = w_{ij} + \alpha . y_j . x_i / \sqrt{(\Sigma_k (w_{kj} + \alpha . y_j . x_k)^2)} - w_{ij}$$

d. Least Mean Squared (LMS):

This model exemplifies classical conditioning and effectively simulates Pavlov's learning process by linking an Unconditional Stimulus (US) with a Conditional Stimulus (CS). A significant correlation between these two elements indicates that the CS can predict the US. Initially, the system does not forecast the US; however, over time, conditioning enables the CS to anticipate the US. During a trial, the predicted value of the US is computed by aggregating the associative values of all present CS. One notable feature of this model is that a stimulus can possess either positive or negative associative values. Positive values serve as conditioned exciters, while negative values act as conditioned inhibitors. Nevertheless, a single stimulus cannot display both types of values simultaneously. The output neuron's activation is calculated as a linear sum of the products of inputs and weights, which underlines the complexity of this process.

Consequently, the activation function is represented as the summation of the input-weight products. The objective is to adjust the network's weights to minimize the error, which is characterized as the disparity between the predicted and ideal outputs [21].

2. Interfaces

a. Coeos:

This is the software component which manages the graphical user interface of the neural network.

b. Simulator-Promethe:

Promethe (developed by Neurocybernetics at the ETIS Lab) is a sophisticated neural network simulator tailored for Unix and MacOS systems. It executes neural scripts, which consist of interconnected "boxes"—each box is defined by a C file that contains various functions.

c. Debugging-Pandora:

Pandora functions as an interface to Promethe (a system designed for simulating neural networks) and it also controls robots or various simulations. There are five categories within this interface; however, the two primary ones are the tab menu and the architecture area. The tab menu contains buttons that are grouped by specific themes—such as Network, General and Save. The Network section, which is quite comprehensive, provides tools for analyzing script performance, but the General section displays the execution time for each box. This functionality assists in identifying performance bottlenecks. Furthermore, the Save feature enables users to save and convert box activity into different formats, which can be particularly useful for analysis.

In addition, the architectural interface (which is quite sophisticated) automatically showcases scripts, each characterized by a distinct color that illustrates essential relationships between various boxes. Each box serves to represent a group, a name and a specific purpose. The left panel of this architectural region features a refresh mode; it allows users to tailor the display of neuron groups according to their preferences. The auto-refresh functionality modifies the display frequency based on a refresh cursor, thus leading to graphical updates without the necessity for data to be refreshed at the same rate. When users right-click within the architecture section, they can access content in frames, thereby enabling them to edit groups—neurons included. Neurons can be altered individually, however, only one neuron may be selected at any given time. Moreover, double-clicking a neuron results in the deletion of its connection.

7.3 Integration

The integration of hardware components with the bio-inspired Neural Network Model is achieved through a Linux-based machine and the Pololu Servo controller , which acts as the primary connecting interface between the hardware and the Bioinspired Neural Network Model. This system facilitates a central communication for all the sensors, actuators and the Unix-based platform that hosts the neural network.

1. **Central Control Unit: Linux-Based Machine.** The Linux-based machine functions as the bridge between the tangible hardware and the software that embodies the neural network. It plays a vital role in processing data from external devices; ensuring the neural network can communicate efficiently with the hardware.

2. **Neural Network Deployment:** deployment occurs on a Linux-based system. This deployment not only enables the establishment of communication via its input/output ports. It also enhances the capacity for real-time data exchange between the network and the connected hardware and also provides more resources for neural network processing.

3. **The hardware components:** the Linux-based machine interfaces with three essential elements: the PAL camera, the Pololu servo controllers, and the actuators. These components create the input and output mechanisms that interact seamlessly with the neural network, although their coordination can be complex.
    a. **PAL Camera:** The PAL camera is directly connected to the Linux-based machine as shown in Fig 17. It captures facial expressions and various elements, this data is then relayed to the neural network through the Linux system, enabling the neural network to process (and analyze) the input for recognition of patterns.

    b. **Pololu:** Instructions from the system running the Neural Network model are sent to the Pololu to make sure the servo motors do the required motions.

    c. **Actuator:** just like the servo controller, the actuator receives movement instructions from the neural network model. These signals are transmitted through the Linux system, which effectively translates the commands to guarantee that the actuator executes the correct movements. This facilitates precise control over mechanical operations.

8. Testing Plan & Result Analysis

8.1 Offline Experiment

We first trained our model offline, with static images, using a balanced dataset of **30 images**, each labeled with one of three gaze directions: **left**, **middle**, or **right** (**10 images per direction**). The training focused on extracting critical facial features using salient points, which were dynamically adjusted based on exclusion file configurations to optimize model performance. Over time, the model exhibited continuous learning, refining its accuracy through iterative adjustments and enhanced feature extraction.

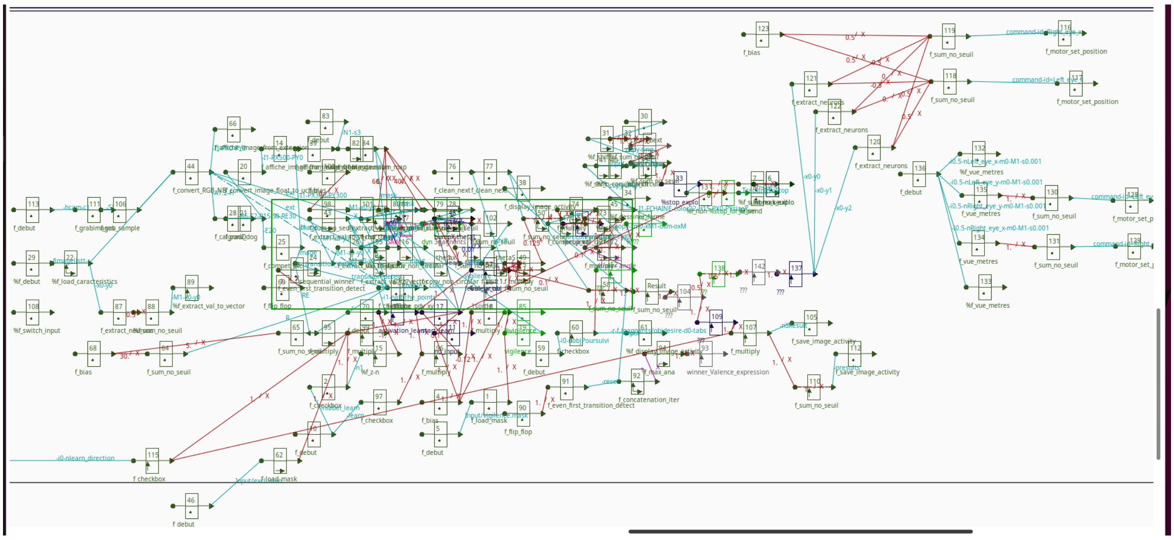


Fig. 18. Complete Neural Model Connection

Understanding Exclusion Values

Each of the four values in the exclusion configuration adjusts the placement of salient points in specific directions:

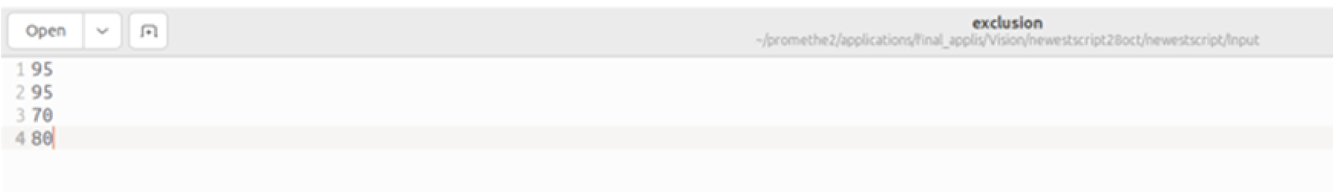


Fig. 19. Exclusion text file containing 4 values

1. **Left Adjustment (First Value):** Controls horizontal placement towards the left. Higher values restrict spread to the left, concentrating points more centrally or to the right.
2. **Right Adjustment (Second Value):** Controls horizontal placement towards the right. Higher values restrict spread to the right, concentrating points more centrally or to the left.
3. **Down Adjustment (Third Value):** Controls vertical placement downward. Higher values restrict spread downward, focusing points more towards the upper areas.
4. **Up Adjustment (Fourth Value):** Controls vertical placement upward. Higher values restrict spread upward, focusing points more towards the lower areas.

These adjustments determine whether salient points are positioned more to the left, right, lower, or upper parts of the face, thereby influencing the model's focus areas.

i. Initial Training Phases

1. **Dataset Composition**: The model was trained on a balanced dataset of 30 images, featuring different individuals. Each image was labeled according to the gaze direction, with 10 images for each direction (left, middle, right).
2. **Initial Feature Extraction**: The model initially utilized 10 salient points, focusing on critical facial features to capture visual cues related to gaze direction.
3. **Training Process**: A supervised learning approach was used, where each set of extracted visual features was associated with a labeled gaze direction.

ii. Testing Phases and Results

The model was evaluated on three datasets to assess its performance and the effect of salient point adjustments on accuracy.

1. Testing on the same dataset

- **Dataset**: The model was tested on the same 30 images used for training.
- **Evaluation Criteria**: Python code was written to analyze SResults.SAVE, calculating correct and incorrect predictions for each gaze direction.
- **Results**:
  - **With 10 Salient Points:**
    - **Exclusion (95, 95, 70, 80)**: **90% Accuracy**

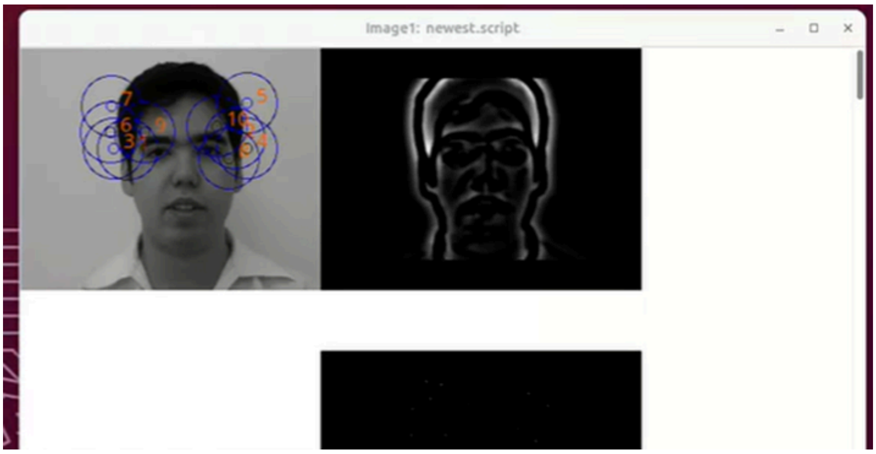


Fig. 20. Training with 10 Salient points

We then changed the exclusion to the following:

- **Exclusion (60, 60, 50, 50)**: **100% Accuracy**

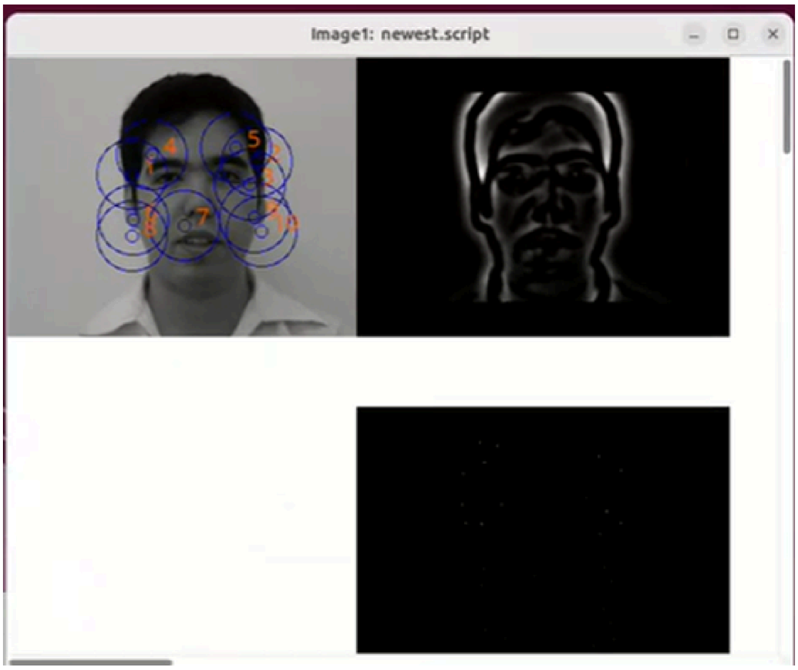


Fig. 21. Training with 10 Salient points with changed exclusion

Figure 20 shows training with 10 salient points, where points are well-placed on key facial features, ensuring accurate gaze classification. In Fig. 21, changed exclusion values shifted the points to cover a more wider range, which included the shoulders in most of the test cases. This clearly made it easier for our model to classify the head direction, as we can tell by the accuracy that it greatly impacted the testing accuracy to achieve a 100%.

We then trained and tested our model with the same dataset, but increased the salient points taken by the model to 20 salient, rather than 10.

- **With 20 Salient Points**:
    - **Exclusion (95, 95, 70, 80)**: **96.7% Accuracy**

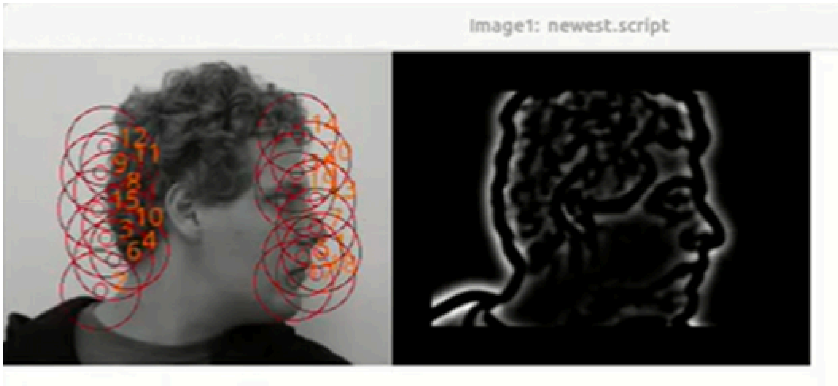


Fig. 22. Training with 20 Salient points (original exclusion)

- **Exclusion (60, 60, 50, 50)**: **100% Accuracy**

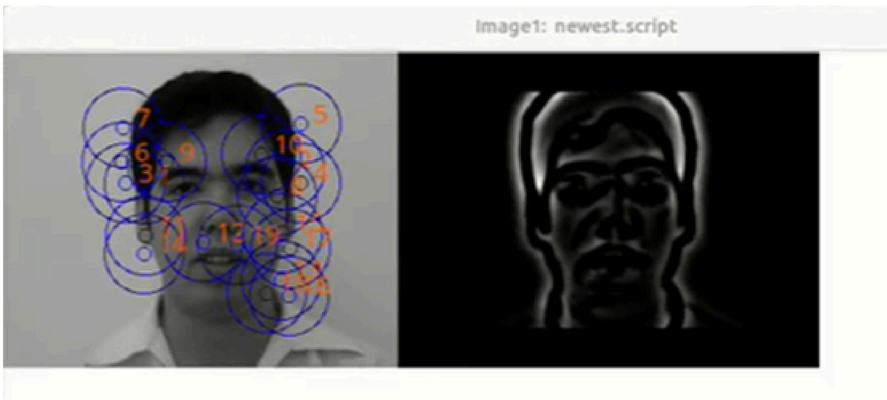


Fig. 23. Training with 20 Salient points with changed exclusion

Therefore, increasing the number of salient points from 10 to 20 allowed the model to capture more peripheral facial features, resulting in improved accuracy and demonstrating the model’s ability to learn and adapt by focusing on regions with higher informational value. Additionally, when all salient points are displayed in the same color during training and testing, it indicates that the model is operating at peak performance, with high accuracy and consistent feature detection across the dataset. This uniformity in point color reflects the model's stability and precision in identifying gaze directions.

2. Testing on a Larger Dataset (420 Images)

- **Dataset**: A larger set of 420 images was tested, excluding the 30 images from the training dataset.
- **Evaluation Criteria**: Python code was written to analyze SResults.SAVE, calculating correct and incorrect predictions for each gaze direction.
- **With 10 Salient Points:**
  - **Exclusion (95, 95, 70, 80)**: **75.95% Accuracy**
  - **Exclusion (60, 60, 50, 50)**: **83.10% Accuracy**
- **With 20 Salient Points:**
  - **Exclusion (95, 95, 70, 80)**: **82.86% Accuracy**
  - **Exclusion (60, 60, 50, 50)**: **93.10% Accuracy**

Therefore, we can come up with the same conclusion as the first test which is that increasing the number of salient points and optimizing the exclusion values significantly improved the model's performance in recognizing gaze directions. With 10 salient points, the highest accuracy achieved was 83.10%, however, by increasing to 20 salient points, the accuracy improved to 93.10%, demonstrating that a higher number of salient points enhances the model's ability to capture more detailed features, resulting in better predictions. Similarly, Exclusion (60, 60, 50, 50) consistently gave better results than Exclusion (95, 95, 70, 80) for both 10 and 20 salient points. This shows that paying attention to features like the shoulders and the curve from the head to the neck works better than just focusing on facial details.

iii. Second Training Phases

4. **Dataset Composition**: The second training phase included training our model on a balanced dataset of 30 images, featuring 10 unique individuals, each looking at 3 different directions. We then tested on the remaining 420 images, excluding the training images.

- **With 10 Salient Points:**
  - **Exclusion (95, 95, 70, 80)**: **79.76% Accuracy**
  - **Exclusion (60, 60, 50, 50)**: **89.05% Accuracy**
- **With 20 Salient Points:**
  - **Exclusion (95, 95, 70, 80)**: **82.62 Accuracy**
  - **Exclusion (60, 60, 50, 50)**: **92.62% Accuracy**

By obtaining the above results, we can conclude that training the model with a dataset of 10 unique individuals to a dataset or a random mix of people, does not make much of a difference in accuracy. This shows that the model is really good at generalizing and doesn't rely on seeing specific individuals during training to perform well. It's able to pick up patterns that work across different people, making it versatile and reliable.

Salient Points and Gaze Direction Mapping

The `f_extract_val_to_vector` function interprets gaze direction based on the arrangement of salient points. The two key functions required for our testing are `f_extract_val_to_vector(89)`, this function extracts the information available and creates a vector label of the true output for each data, and the `f_sum_no_seil(Result)` function shows the final result (prediction) the model was able to come up with. The visualization of the function, `f_sum_no_seil(result)` shows how confident the model is in its predictions and the direction the model thinks the subject is looking. The area of the row shows us how confident the model is with its answer.

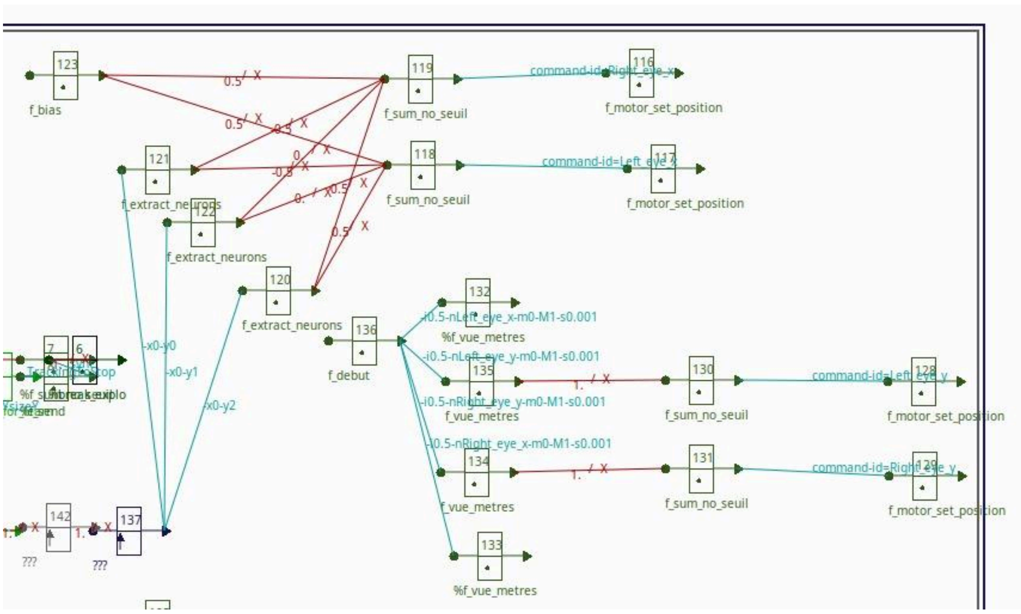


Fig. 24. Neural model connection for 3 distinct direction

- **First Row (White bar in the first row):** Gaze direction is towards the middle.
- **Second Row (White bar in the second row):** Gaze direction is towards the left.
- **Third Row (White bar on the third row):** Gaze direction is towards the right.

Fig. 25. Sample of actual (left bar) vs predicted (right bar) value of the model for the middle direction

This mapping ensures that the model accurately interprets the spatial distribution of salient points to classify gaze direction effectively.

Example of Correct Testing Visuals

- **Person Looking Middle:**

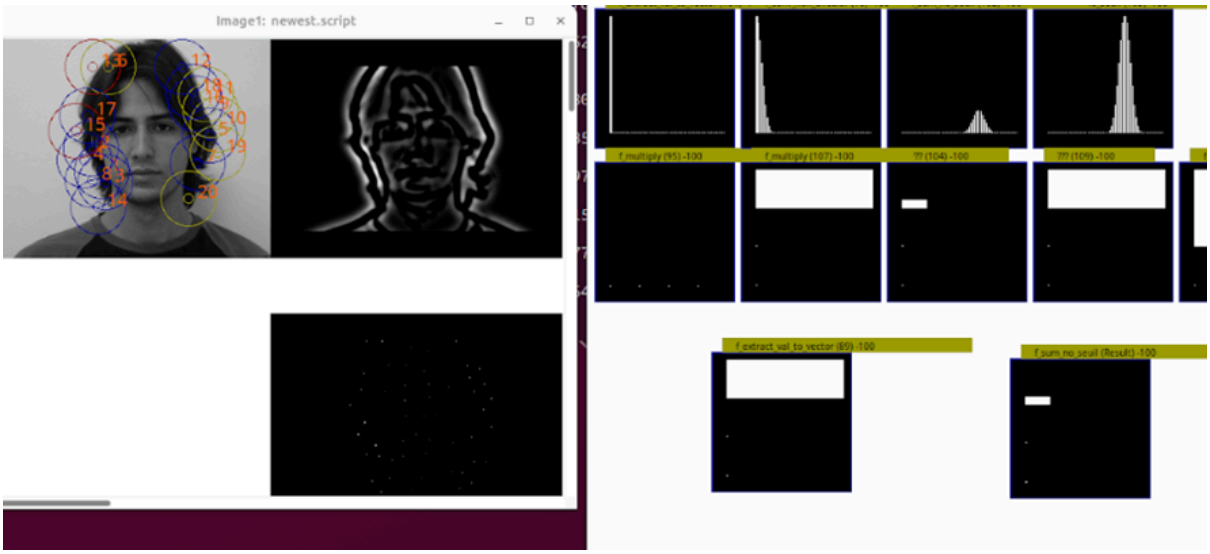


Fig. 26. Correct prediction of a person looking towards the middle

- **Person Looking Left:**

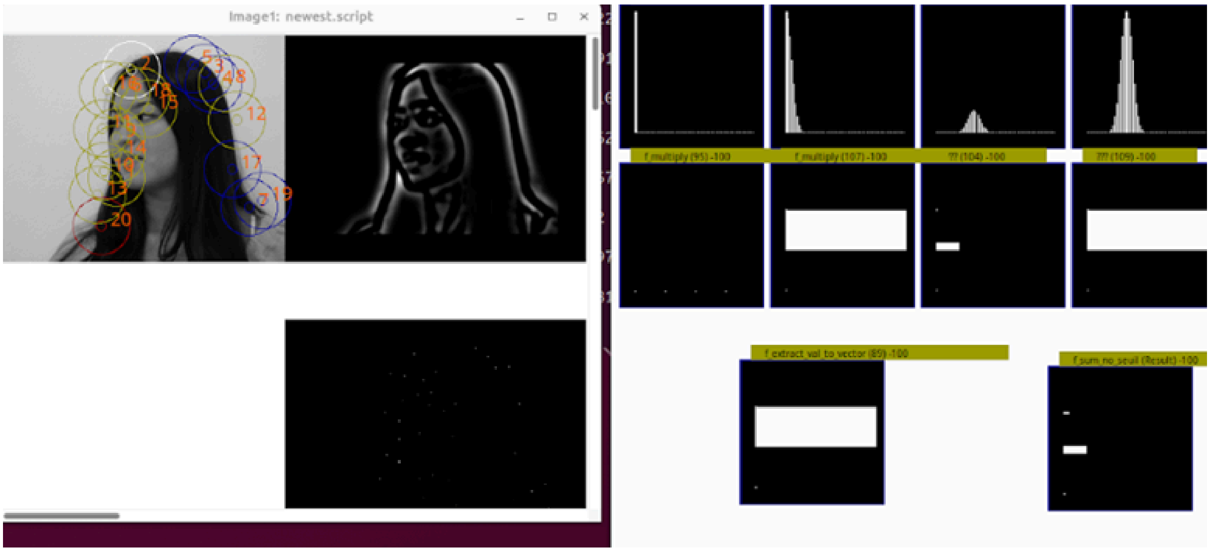


Fig. 27. Correct prediction of a person looking towards the left

- **Person Looking Right:**

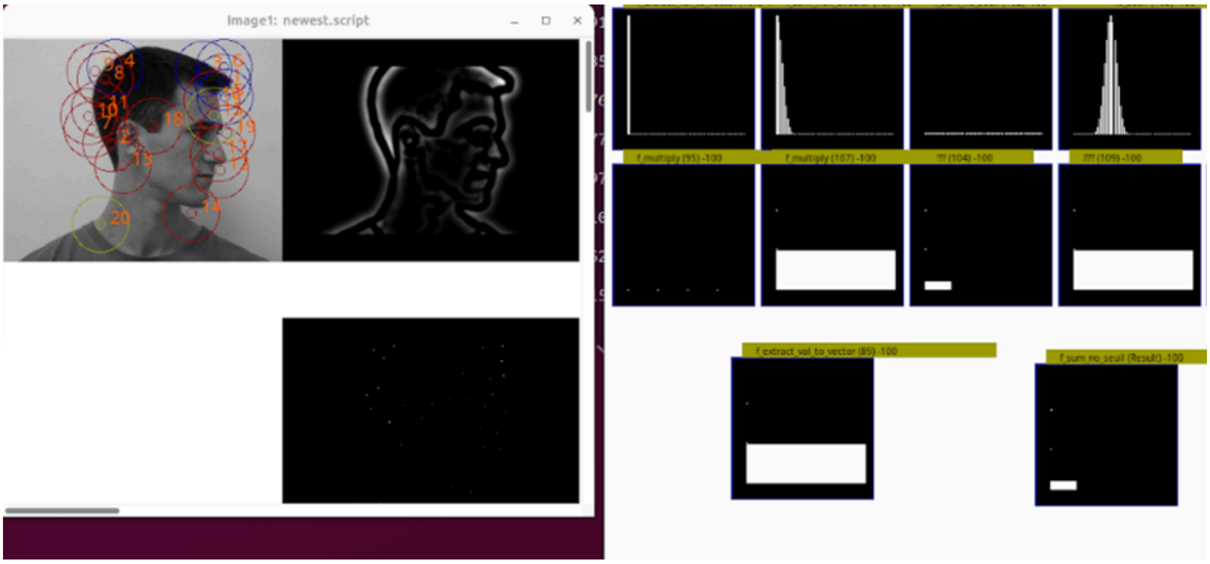


Fig. 28. Correct prediction of a person looking towards the right

Summary of Testing Results

- **Initial Training and Testing (30 images of random people)**
    1. **Testing Set (30 Images):**

Table 2. Testing on the same Training Data (random individuals training)

Salient Points	Exclusion (95,95,70,80)	Exclusion (60,60,50,50)

10 Points	90%	100%
20 Points	96.7%	100%

2. **Testing Set (420 Images):**

Table 3. Testing on 420 images not seen before (random individuals training)

Salient Points	Exclusion (95,95,70,80)	Exclusion (60,60,50,50)
10 Points	75.95%	83.10%
20 Points	82.86%	93.10%

- **Second Training (30 images of 10 unique individuals)**
  1. **Testing Set (420 Images):**

Table 4. Testing on 420 images not seen before (unique individuals training)

Salient Points	Exclusion (95,95,70,80)	Exclusion (60,60,50,50)
10 Points	79.76%	89.05%
20 Points	82.62%	92.62%

The offline training and testing phases demonstrated that the model's accuracy is significantly enhanced through optimized salient point placement and strategic exclusion configurations. The dynamic adjustment of salient points, represented by small colored circles, highlights the model's ability to learn and adapt efficiently. This aligns with the principles of bio-inspired learning, where models are designed to achieve high performance by training on small datasets and effectively generalizing to much larger ones, showcasing their robustness and adaptability in real-world applications.

iv. Final Training Phase

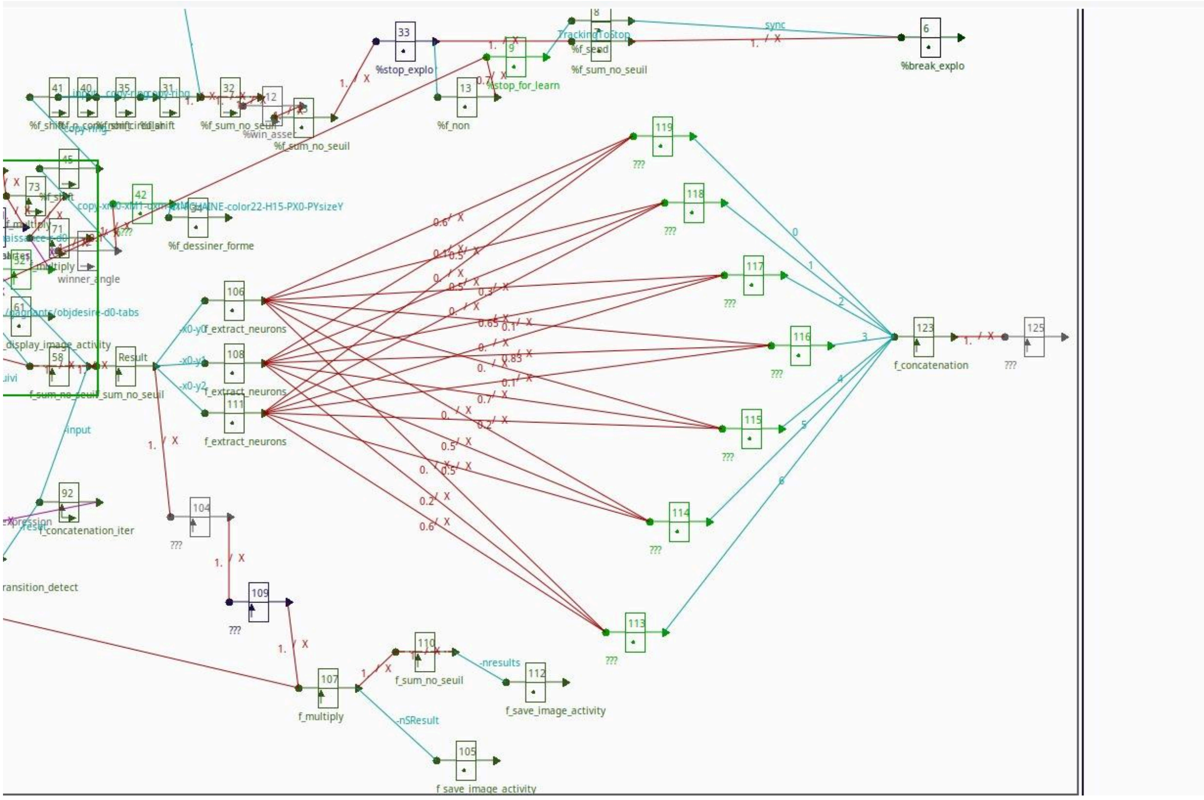


Fig. 29. Neural model with 7 positions for direction

The final phase of the offline experiment focused on enhancing our bio-inspired neural network by expanding its capability to classify gaze directions into **seven categories**: extreme right, moderate right, slight right, middle, slight left, moderate left, and extreme left. To achieve this, we restructured the output layer to include seven neurons arranged vertically, where the top neuron represents extreme right, followed by moderate right, slight right, middle, slight left, moderate left, and finally extreme left at the bottom. This design aimed to improve the model's ability to capture subtle variations in gaze direction, addressing real-world scenarios where these differences are significant. Additionally, a larger testing dataset was introduced to assess the model's generalizability and robustness under more complex conditions.

- **Training Dataset:** 30 images of random individuals, previously labeled for three primary gaze directions (left, middle, right).
- **Model Architecture:** The output layer was expanded to include 7 neurons, each representing a distinct gaze direction: extreme right, moderate right, slight right, middle, slight left, moderate left, and extreme left.

Training Process

The supervised learning process involved associating the extracted salient features with one of the seven gaze directions. The team refined the model's weights to minimize prediction errors, leveraging its ability to detect both central and peripheral features.

v. Final Testing Phase

Dataset

- **Testing Dataset:** A new set of 254 images featuring individuals not present in the training dataset.
- The dataset included variations in facial angles, and head positioning to evaluate the model's adaptability.

Evaluation Criteria

1. **Correct but not accurate predictions:**
    - Any gaze classified within left (extreme, moderate, slight) or right (extreme, moderate, slight) was considered correct, as even humans struggle to distinguish these subtle variations.
    - For middle gaze, predictions of slight right or slight left were counted as correct due to minimal deviation.
2. **Incorrect predictions:**
    - Left variations classified as right, or vice versa.
    - Middle classified as extreme or moderate left/right.

Results

- **Overall Accuracy:** 96.85%.

Examples of Model Predictions

- **Correctly Predicted Images:**

The model successfully classified most images, even in challenging scenarios where the gaze direction was borderline between categories.

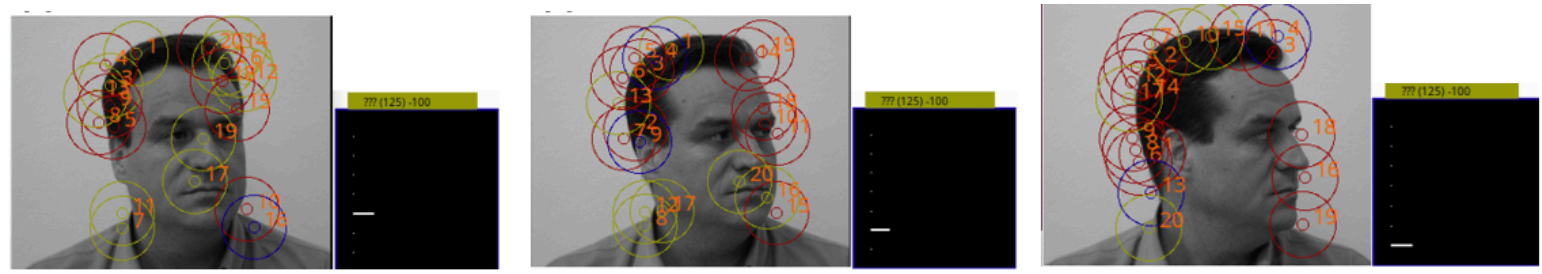


Fig. 30. Example of a correctly predicted slight, moderate, and extreme right, respectively

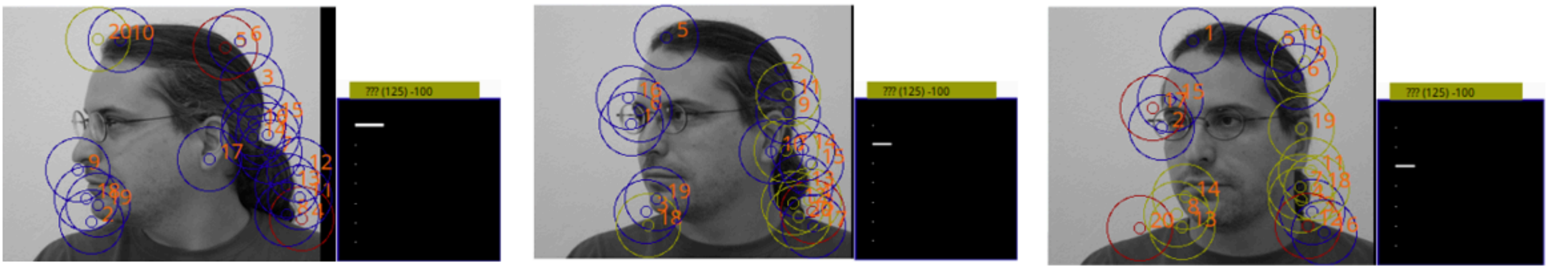


Fig. 31. Example of a correctly predicted slight, moderate, and extreme left, respectively

As you can see from the images, distinguishing between slight, moderate, and extreme head variations for each gaze direction is inherently challenging. Even for humans, subtle differences in head angles and gaze shifts can be difficult to classify consistently. This highlights the complexity of the task and underscores the model's ability to achieve high accuracy despite the nuanced nature of the dataset.

- **Incorrectly Predicted Images:**

Misclassifications were rare but notable, often involving middle gazes classified as extreme deviations or left variations classified as right.

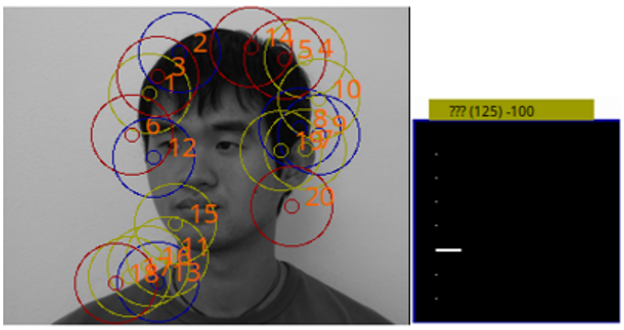


Fig. 32. Example of a slight left image incorrectly predicted as slight right

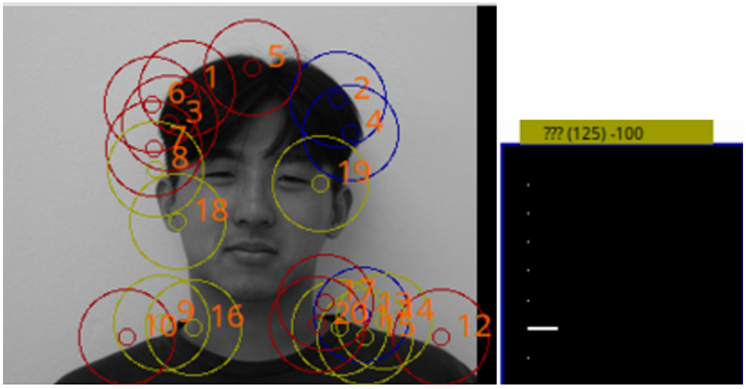


Fig. 33. Example of a middle image incorrectly predicted as moderate right

Analysis and Findings

The inclusion of 7 neurons allowed the model to handle subtle gaze variations effectively. Despite the increased complexity, the model maintained high accuracy across the testing dataset.

The evaluation criteria accounted for human-like limitations, acknowledging that certain classifications, such as slight left vs. moderate left, may inherently have a degree of ambiguity.

The addition of seven output neurons improved the model's capability to classify gaze directions with high accuracy. The model's overall performance reflects its adaptability and precision given the rare cases it struggled with extreme misclassifications.

8.2 Real-Time Experiment

Learning Phase (Online Learning Mode)

In this phase, the system was tested on human subjects in real time, learning directly from their gaze behaviors. The robot was first trained on all team members. So, every member sat opposite of the robot and looked towards a random direction (left, middle, or right). The robot then would cycle through the three internal states of direction it has, and the subject would try and replicate them. The robot would learn by extracting the visual features as saline points and associating them to the direction the eye has moved to. The robot then learned the different directions from all team members and was later tested to view its accuracy.

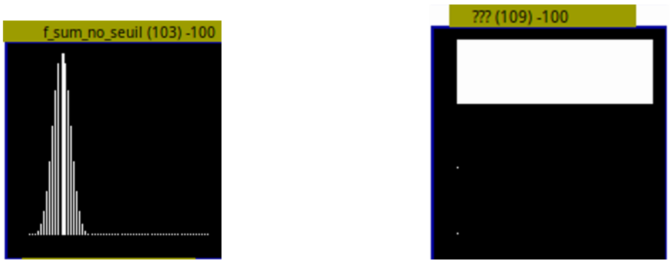


Fig. 34. Graph for subject facing left

Similar to offline testing the function f_sum_no_seil shows the final result the model was able to come up with. The visualization of the function is represented by the two graphs 103 and 109. Graph 103 is key to our research as it illustrates how the bio-inspired model has higher firing rate at specific neurons similar to the head cell found in organic beings.

f_sum_no_seil-(109) show how confident the model is in its predictions depending on the size of the row and the direction the model thinks the subject is looking. The area of the row shows us how confident the model is with its answer. The true labels can only be visually verified to see if the model is working as the model is trying to extract features at real-time of the subject through the camera that is embedded in the eyes of the robot.

The mapping we found through visual inspection the model is using tor represent is as follows:

- **Gaze direction of the robot towards the Left:** The White bar in the first row in Graph 109 shows that robot interprets this row as Gaze direction is towards the left. There is also high activity in the left section in the 103 Graph as seen in the figure below. third point is that we more blue saline points when the subject is facing towards the left.

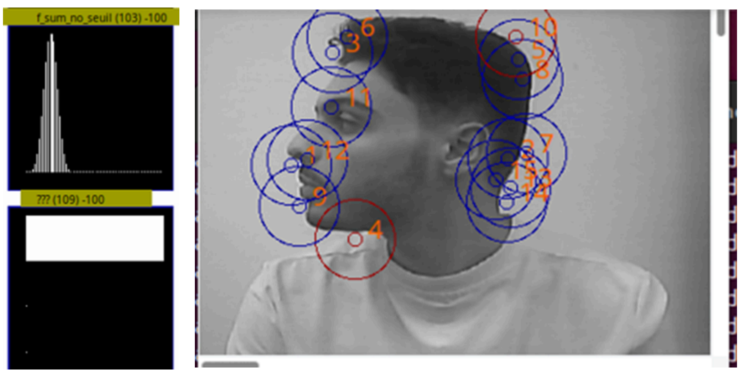


Fig. 35. Example of real time learning for left direction

- **Gaze direction of the robot towards the Middle:** The White bar in the second row in Graph 109 shows that the robot interprets this row as Gaze direction is towards the middle. There is also high activity in the middle section in the 103 Graph as seen in the figure below. The third point is that we have more yellow saline points when the subject is facing towards the middle.

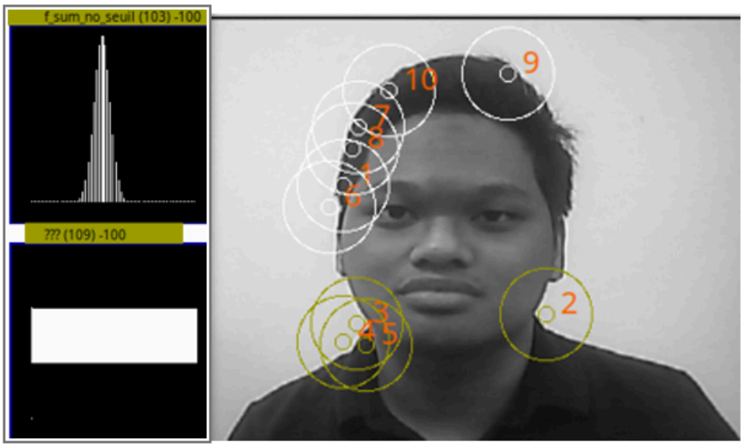


Fig. 36. Example of real time learning for central direction

- **Gaze direction of the robot towards the Right:** The White bar in the last row in Graph 109 shows that the robot interprets this row as Gaze direction is towards the right. There is also high activity in the right section in the 103 Graph as seen in the figure below. the third point is that we have more red saline points when the subject is facing towards the right.

Fig. 37. Example of real time learning for right direction

Test Phase (Use Mode)

In this test phase, the team checked the accuracy of the model in replicating human gaze in the real world. The neural network uses the live footage of the person's head and eye movements captured by the robot's camera to estimate the direction of gaze. Natural interaction is therefore made possible by the robotic head and eyes mimicking these actions. Accuracy, response time, and flexibility in different situations, such as changing lighting or user distance, were important factors that contributed to the overall accuracy of the system. The model was further improved through iterative modifications to guarantee a dependable performance.

9. Validation, Verification, and Performance Analysis Plan

The main objective of studying human-robot interaction in our robot prototype is to imitate the human gaze with a camera installed on a robotic neck. One servo motor will be attached to the neck, while the other three servos will be employed to move the robot's head in a way that imitates human movement. The robot's servo motors will imitate the movements of the head and eyes. Our goal is to replicate the way humans look at things by using a biomimetic neural network model, ensuring that our software and hardware components are correctly implemented through the use of both white-box and black-box testing.

The primary goal of the validation process is to employ a step-by-step process that instructs the robot to mimic the human gaze. Every link is crucial as the neural network enhances its comprehension of human eye movements. Experimental studies involving human participants confirm the efficacy of the bioinspired neural network for gaze estimation. Calibration processes will be performed to verify the correlation between neural network values and gaze directions, guaranteeing precise and dependable gaze estimation in various individuals and environmental settings.

The individuals engaging with the robot act as the evaluators, verifying the accuracy of the robot's neck and eye mimicry. This repetitive verification process of the brain's function stores the information for future eye movements. Accurate identification of gaze direction through systematic data collection by capturing images and videos of each team member's face is essential for the success of our learning process.

The accuracy of our testing and aligning the robot's gaze direction with the users' expressions are crucial for our success. The team aims for precision in individual tests. Nevertheless, in the case of generalization, the team strives for an adequate level of accuracy when evaluating random individuals. These series of experiments aim to further investigate human-robot interactions and showcase the adaptability and learning capabilities of robotic systems in social settings.

Experimental studies involving human participants validate the effectiveness of bioinspired gaze estimation. Calibration procedures are carried out to confirm the relationship between neural network values and gaze directions, guaranteeing precise and dependable gaze estimation in various individuals and environmental circumstances. The team will also evaluate the robustness of the model to factors such as changes in lighting conditions, background noise, and variations in head posture. This evaluation ensures the model remains effective and reliable in different environments and scenarios.

10. Project Global, Economic, and Societal Impact

A. Global Impact

The progress of robotics is leading the way in technological development, ready to transform industries and change the way humans and machines interact. Integrating bio-inspired neural networks into robotics technology has the potential to drive this advancement to new levels. Engineers can create robots with cognitive abilities similar to humans by studying how the human brain works. These neural networks inspired by biology allow robots to understand, acquire knowledge, and adjust to their surroundings in ways that go beyond conventional programming approaches. Therefore, robots with these neural architectures can engage in more complex, subtle interactions with humans, allowing for smooth collaboration across various fields. The incorporation of bio-inspired neural networks into a variety of fields, from healthcare and manufacturing to service industries and more, marks a future where machines not only help but also interact with humans in natural and intuitive ways, opening up new realms of innovation and potential.

B. Economic Impact

Post-development of the bioinspired gaze-direction learning technology may provide opportunities to introduce new job openings. Moreover, since the project calls for the need to employ people trained in research and development, the manufacturing of the gaze-direction learning robot and the specialists responsible for the maintenance of the TurtleBot3 model are highly required whenever the system is demanded for regular checkups. These fields are necessary to be comparable to the robot with the gaze-direction technology to be functional when socially interacting with human users. However, some existing jobs might reduce the need for human employees; displacing workers such as receptionists with robots encoded with gaze-direction learning. Potentially, the expected growth of robotics that depends on visual cues would significantly rise to an 8.5% growth rate and exceed $61.4 billion by 2025 [17]. Multiple industries utilize more robotics in their respective sectors, including healthcare, entertainment, education, customer services, and other zones that employ robotics.

C. Societal Impact

By mimicking the human gaze behavior, robots can interact more naturally with humans. Encouraging natural interactions between humans and robots promotes acceptance and integration of robots and AI in the daily lives of people, therefore increasing their effectiveness in assisting humans. Moreover, natural human-robot interaction fosters trust and collaboration which pave the way for the seamless integration of robots in a social setup.

As the field of robotics continues to advance toward making robot behavior more similar to that of humans, such changes invoke many important considerations concerning its potential application, such as ethical issues, and the effect it has on society. Some of the most striking concerns are tied to privacy issues, personal rights, and the principle of fairness. For example, a robot that can reproduce human-like gaze can also collect and utilize personal data on the person under its gaze. Another important consideration is how the use of robots in daily activities should be regulated. There are also broader ethical questions that should be considered, such as the reciprocity and delegation of decision-making power from one human to a robotic system. Such questions may have ethical overtones that extend to the potential for even further labor displacement in some industries and the just distribution of the advancements in robotics technology. As a result, it is vital to consider these ethical issues so that the gaze-mimicking robots' growth and deployment are ethically and socially acceptable.

11. Standards

The project is committed to adhering to established standards in hardware, software, and ethical practices to ensure the development of responsible robotic systems. The manufacturing of hardware strictly follows industry protocols to ensure the physical safety and reliability of the 3D-printed robot head. Incorporating sensory integration technology enhances the system's nonverbal communication abilities, maintaining compatibility with older standards. Safety in the circuit is further ensured through the use of a relay. Ethical research guidelines, including informed consent, participant confidentiality, and the protection of individuals involved, are rigorously followed. The project also upholds recognized data collection and validation protocols, ensuring a systematic and transparent approach. Through its commitment to these comprehensive standards, the project strives to contribute responsibly to human-robot interaction, emphasizing safety, ethics, and reliable results for broader scientific application.

12. Conclusion

Our goal was to enhance social robotics by creating a gaze-mimicking robot that can replicate human-like gaze behavior. The team was able to accomplish this milestone by developing a robotic system that can naturally and intuitively recognize, process, and react to human attention cues by utilizing bio-inspired neural networks. The team was able to create a neural network that is inspired by the working of head-direction cells found in living organisms. The model was able to successfully mimic seven distinct horizontal eye gaze directions with high accuracy after initial testing. The robot was able to recognize and mimic the human gaze direction by extracting essential features for recognition during the training process. The robot achieved almost perfect accuracy with right saliency exclusion tuning.

Notably this research showed us that bio-inspired neural networks are able to perform better than traditional models when learning with a small amount of data points. By merging theoretical Neuroscience understandings with practical Engineering solutions, the team tried to lead a new path in human-robot interaction advancements and set the stage for deeper and more captivating interactions between humans and robots. The team believes that the gaze-mimicking robot using a bio-inspired neural model will not only demonstrate the benefits of interdisciplinary research but will also aid in the progress of social robotics and the development of more empathetic and responsive robotic companions.

References

[1] C. K. Peck, "Neuronal activity related to head and eye movements in cat superior colliculus.," The Journal of Physiology, vol. 421, no. 1, pp. 79–104, Feb. 1990, doi: https://doi.org/10.1113/jphysiol.1990.sp017934.

[2] Ferigo, A. *et al.* (2023) 'Evolving Hebbian learning rules in Voxel-based soft robots', *IEEE Transactions on Cognitive and Developmental Systems*, 15(3), pp. 1536–1546. doi:10.1109/tcds.2022.3226556.

[3] Guestrin, E.D. and Eizenman, M. (2006) 'General theory of remote gaze estimation using the Pupil Center and Corneal Reflections', *IEEE Transactions on Biomedical Engineering*, 53(6), pp. 1124–1133. doi:10.1109/tbme.2005.863952.

[4] Yoo, D.H. and Chung, M.J. (2005) 'A novel non-intrusive eye gaze estimation using cross-ratio under large head motion', *Computer Vision and Image Understanding*, 98(1), pp. 25–51. doi:10.1016/j.cviu.2004.07.011.

[5] M. Q. Khan and S. Lee, "Gaze and Eye Tracking: Techniques and Applications in ADAS," *Sensors*, vol. 19, no. 24, p. 5540, Dec. 2019, doi: https://doi.org/10.3390/s19245540.

[6] W. Wang, "Integrating GazeCloudAPI, a high accuracy webcam based eye-tracking solution, into your own web-app," *Medium*, Nov. 13, 2020. https://medium.com/@williamwang15/integrating-gazecloudapi-a-high-accuracy-webcam-based-eye-tracking-solution-into-your-own-web-app-2d8513bb9865 (accessed Mar. 16, 2024).

[7] Z. Bing *et al*, "A Survey of Robotics Control Based on Learning-Inspired Spiking Neural Networks," *Frontiers in Neurorobotics,* 2018. Available: http://aus.idm.oclc.org/login?url=https://www.proquest.com/scholarly-journals/survey-robotics-control-based-on-learning/docview/2294005174/se-2. DOI: https://doi.org/10.3389/fnbot.2018.00035.

[8] A. Soltoggio *et al*, "Rare Neural Correlations Implement Robotic Conditioning with Delayed Rewards and Disturbances," *Frontiers in Neurorobotics,* 2013. Available: http://aus.idm.oclc.org/login?url=https://www.proquest.com/scholarly-journals/rare-neural-correlations-implement-robotic/docview/2294047457/se-2. DOI: https://doi.org/10.3389/fnbot.2013.00006.

[9] S. Park, D. Kim and B. C. Song, "Fine Gaze Redirection Learning with Gaze Hardness-aware Transformation," *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA, 2023, pp. 3453-3462. Available: https://openaccess.thecvf.com/content/WACV2023/papers/Park_Fine_Gaze_Redirection_Learning_With_Gaze_Hardness-Aware_Transformation_WACV_2023_paper.pdf. DOI: 10.1109/WACV56688.2023.00346

[10] E. G. Freedman, "Coordination of the eyes and head during visual orienting," *Experimental Brain Research*, vol. 190, no. 4, pp. 369–387, Aug. 2008, doi: https://doi.org/10.1007/s00221-008-1504-8.

[11] P. Delarboulas, P. Gaussier, R. Caussy, and M. Girard, "Robustness Study of a Multimodal Compass Inspired from HD-Cells and Dynamic Neural Fields," in *From Animals to Animats 13*, 2014, vol. 8575, ISBN: 978-3-319-08863-1.

[12] P. Gaussier and S. Zrehen, “Perac: A neural architecture to control artificial animals,” *Robot. Auton. Syst.*, vol. 16, no. 2, pp. 291–320, 1995.

[13] S. Amari, “Dynamics of pattern formation in lateral-inhibition type neural fields,” *Biol. Cybern.*, vol. 27, no. 2, pp. 77–87, 1977.

[14] Peyrache, A., Lacroix, M. M., Petersen, P. C., & Buzsáki, G. (2015). Internally organized mechanisms of the head direction sense. Nature neuroscience, 18(4), 569–575. https://doi.org/10.1038/nn.3968

[15] S. Cristina and K. P. Camilleri, "Gaze Tracking by Joint Head and Eye Pose Estimation Under Free Head Movement," *2019 27th European Signal Processing Conference (EUSIPCO)*, A Coruna, Spain, 2019, pp. 1-5, doi: 10.23919/EUSIPCO.2019.8902786.

[16] D. Cazzato, F. Dominio, R. Manduchi, and S. M. Castro, “Real-time gaze estimation via pupil center tracking,” *Paladyn, Journal of Behavioral Robotics*, vol. 9, no. 1, pp. 6–18, Feb. 2018, doi: https://doi.org/10.1515/pjbr-2018-0002.

[17] M. Reshetnikova, and I. Pugacheva. “The Global Industrial Robotics Market: Development Trends and Volume Forecast” *Current Problems of the World Economy and International Trade (Research in Economic Anthropology, Vol. 42)*, Emerald Publishing Limited, Leeds, pp. 187-195, Mar. 2022, doi: https://doi.org/10.1108/S0190-128120220000042018

[18] A. Dević et al, "SYNTHESIS OF THE CONTROL UNIT OF THE DESKTOP ROBOT ARM ACTUATED BY STEPPER MOTORS," Annals of the Faculty of Engineering Hunedoara, vol. 21, (4), pp. 85-88, 2023. Available: http://aus.idm.oclc.org/login?url=https://www.proquest.com/scholarly-journals/synthesis-control-unit-desktop-robot-arm-actuated/docview/2928342475/se-2.

[19] Z. Wang et al, "Bioinspired Stimuli-Responsive Materials for Soft Actuators," Biomimetics, vol. 9, (3), pp. 128, 2024. Available: http://aus.idm.oclc.org/login?url=https://www.proquest.com/scholarly-journals/bioinspired-stimuli-responsive-materials-soft/docview/2997630838/se-2. DOI: https://doi.org/10.3390/biomimetics9030128.

[20] C. Bergmeir, & J. M. Benitez, “Neural Networks in R Using the Stuttgart Neural Network Simulator: RSNNS.” Journal of Statistical Software, 2012, 46(7). https://doaj.org/article/0b6a3e52b406460badc1b535159a8d12.

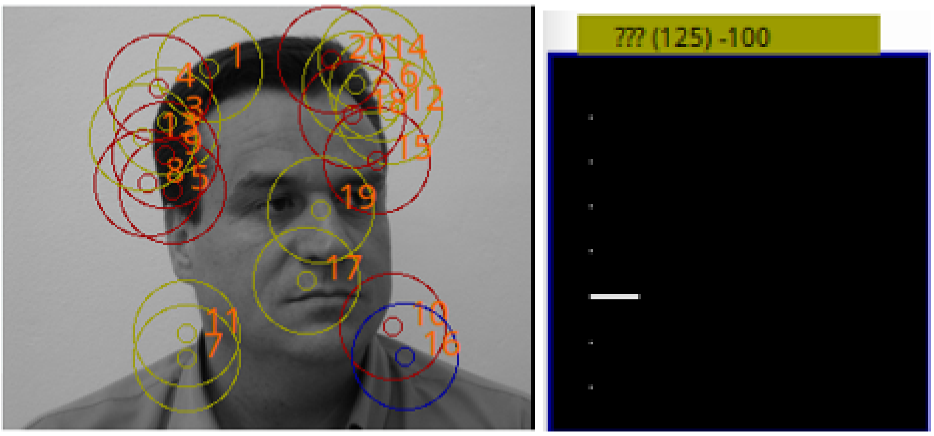
[21] A. Moualla, “Un robot au Mus´ee : Apprentissage cognitif et conduite esth´etique,” thesis, University of Cergy-Pontoise, Cergy, 202AD

Appendix

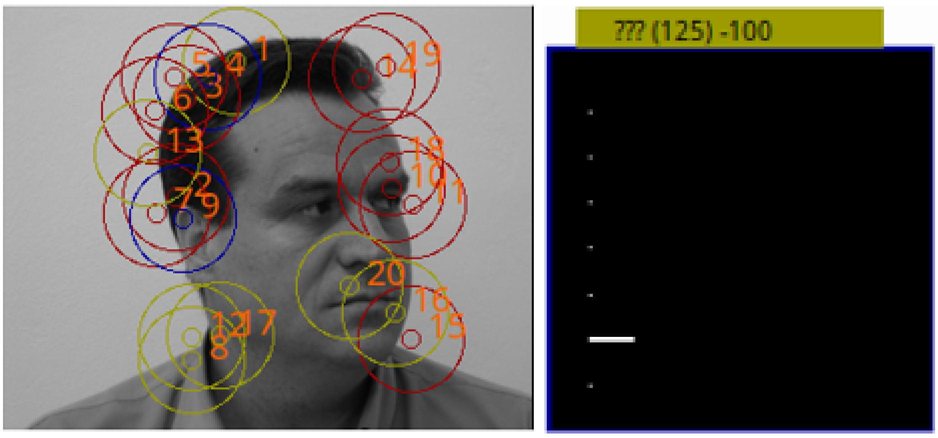
Appendix A - Offline Test Results with 7 Angles

All images mentioned below are the offline test images and their respected results from our model; 254 images were evaluated.

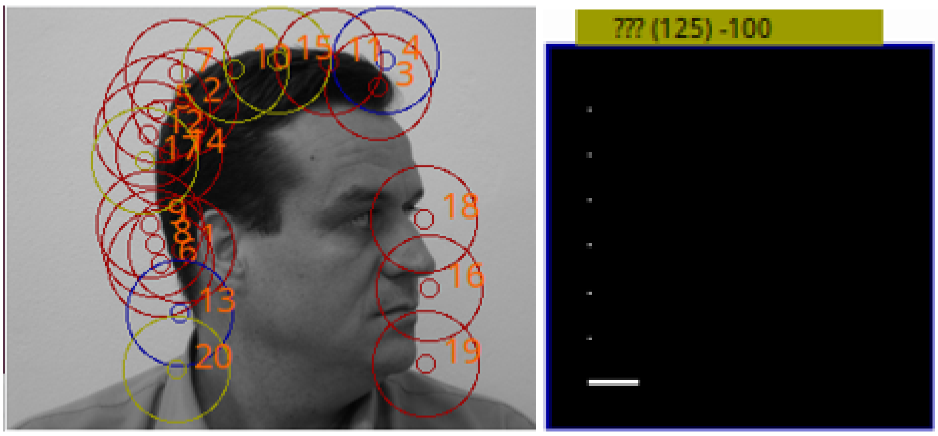
1.png



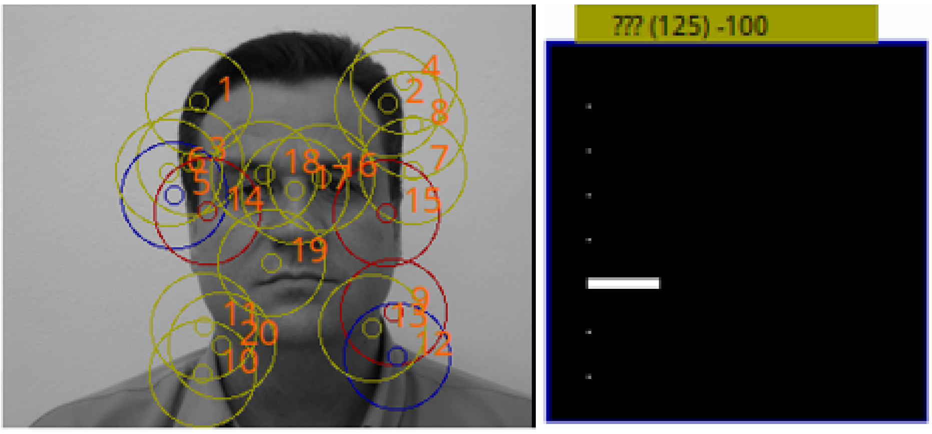
2.png



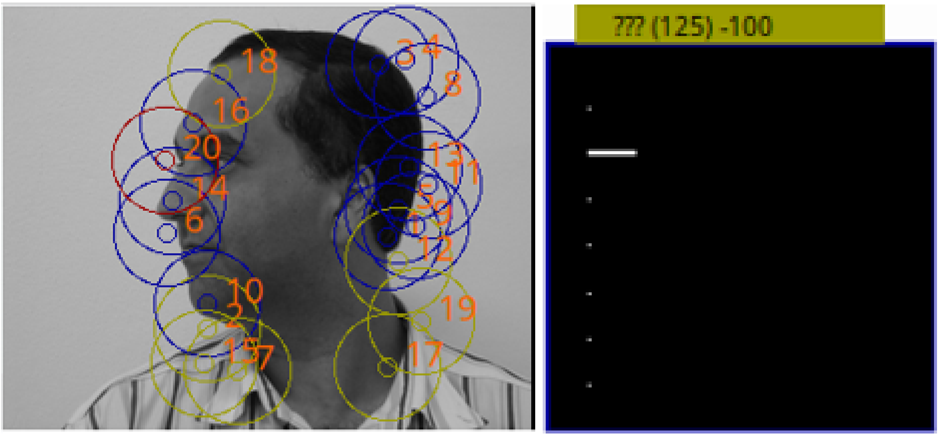
3.png



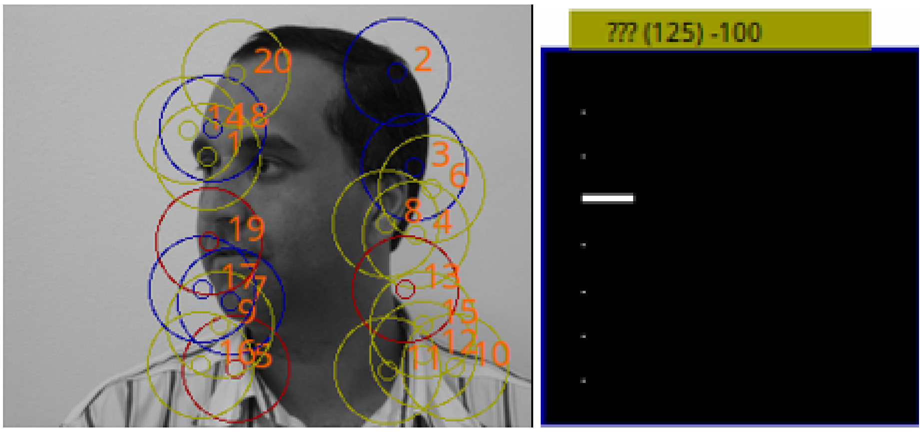
4.png



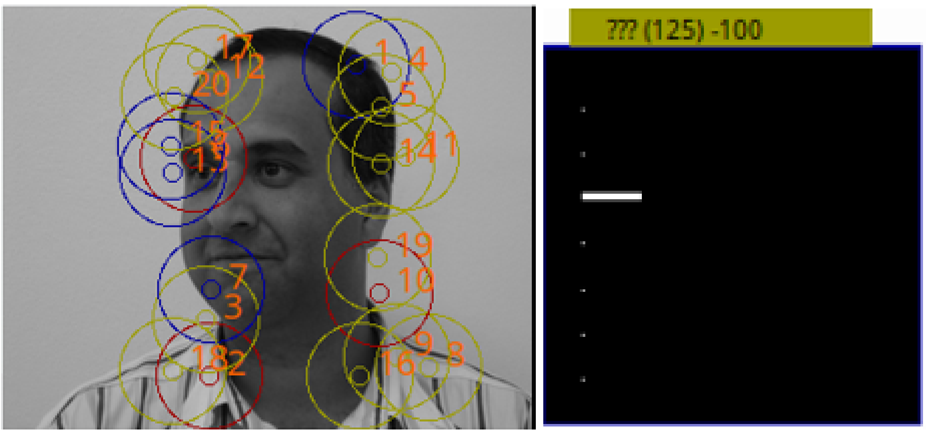
5.png



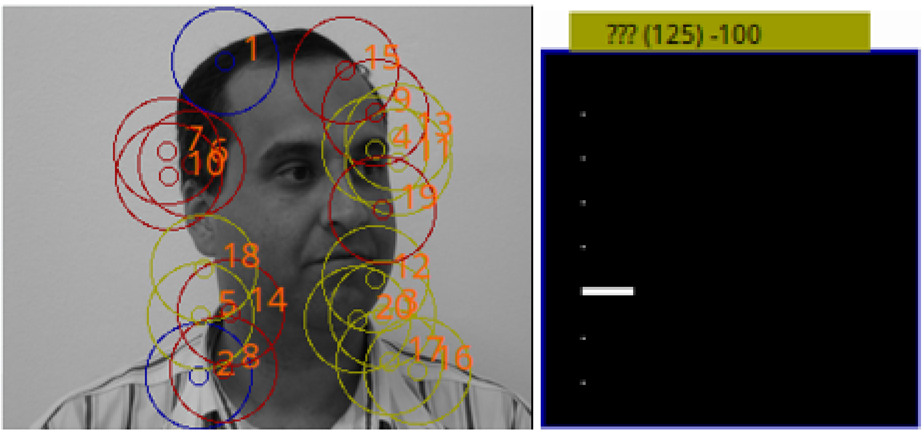
6.png



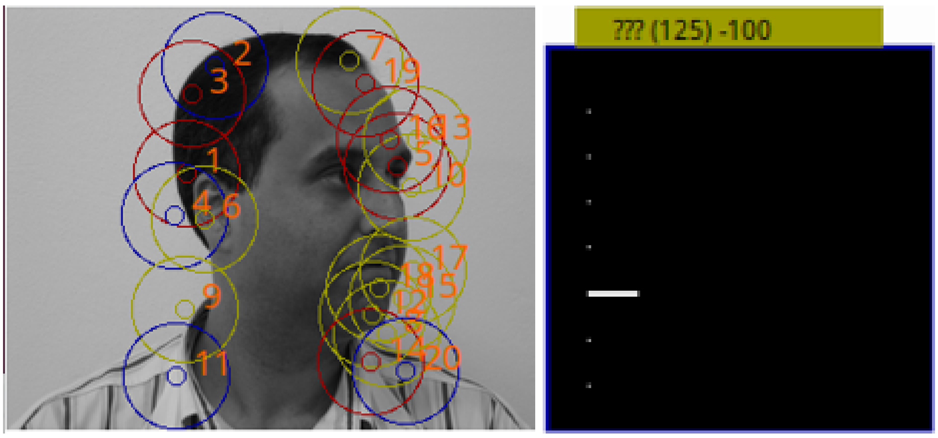
7.png



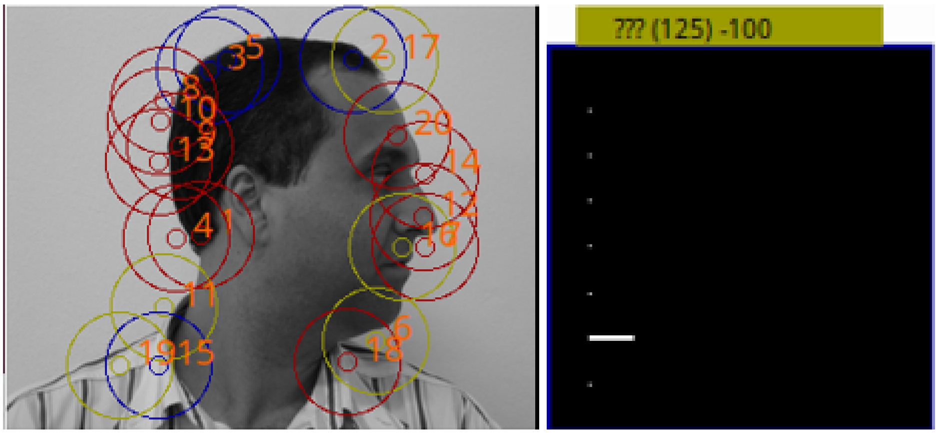
8.png



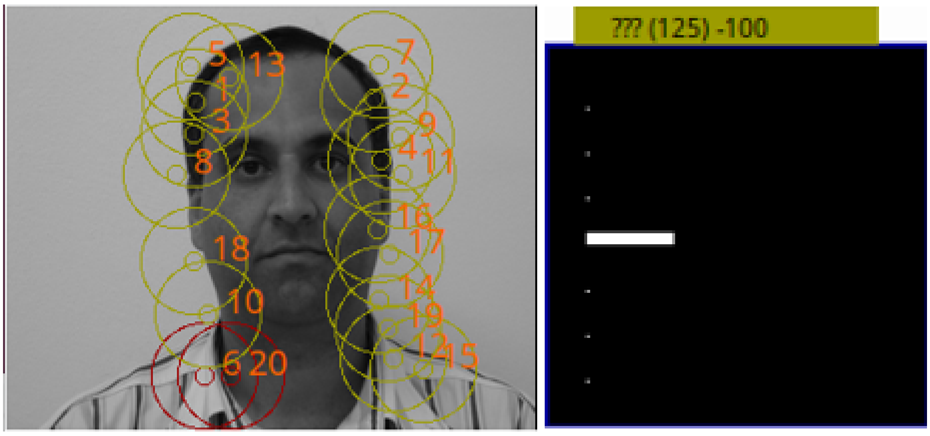
9.png



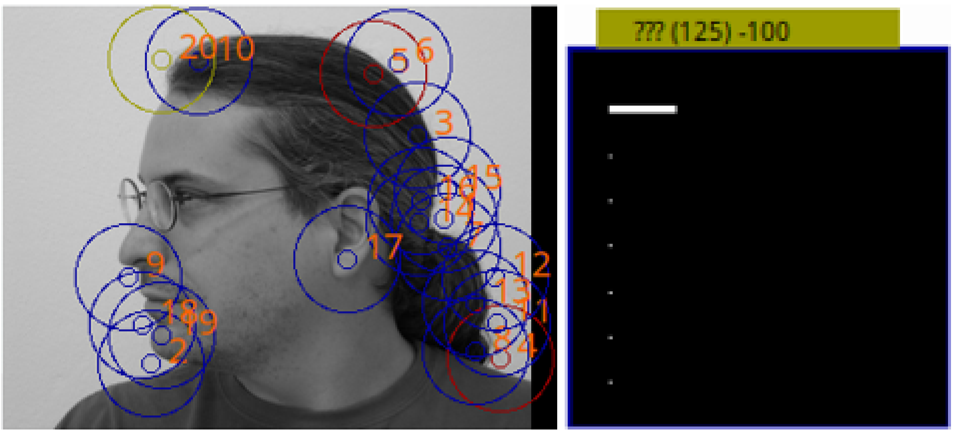
10.png



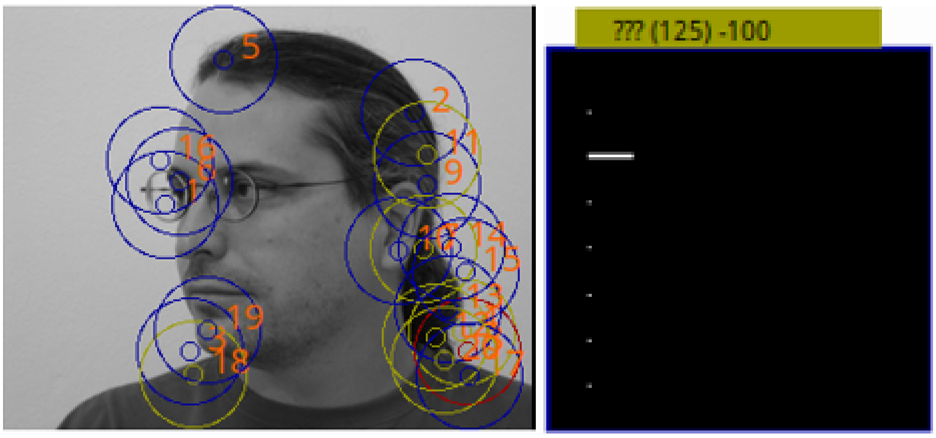
11.png



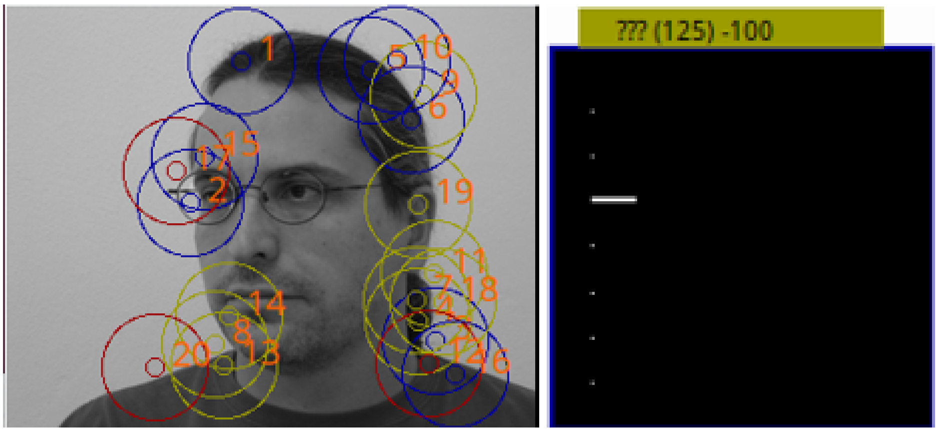
12.png



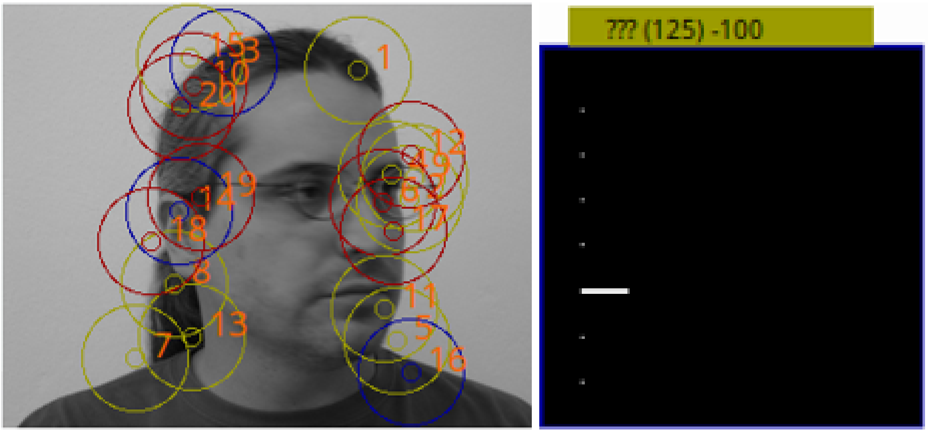
13.png



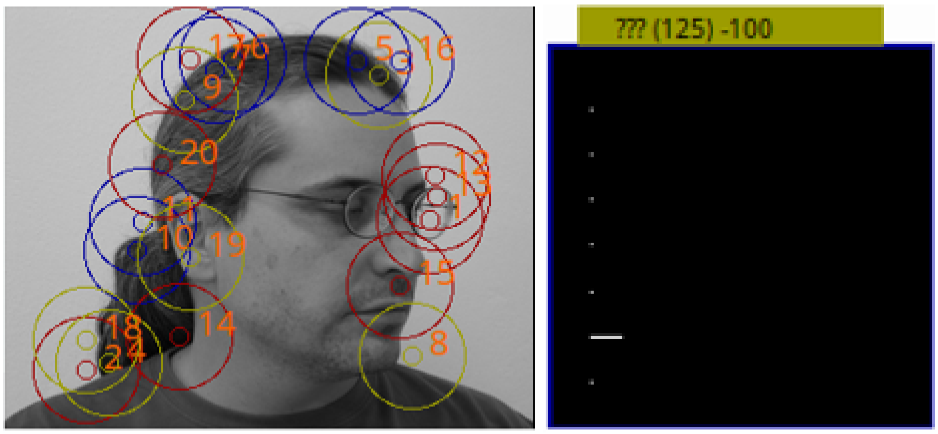
14.png



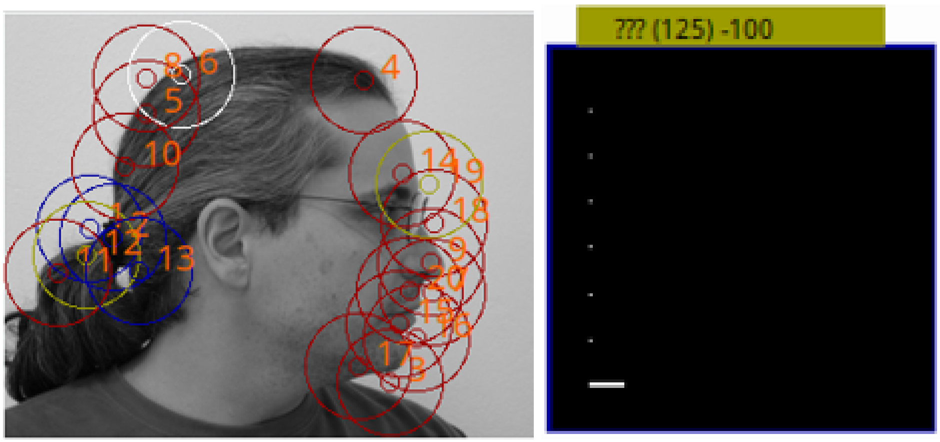
15.png



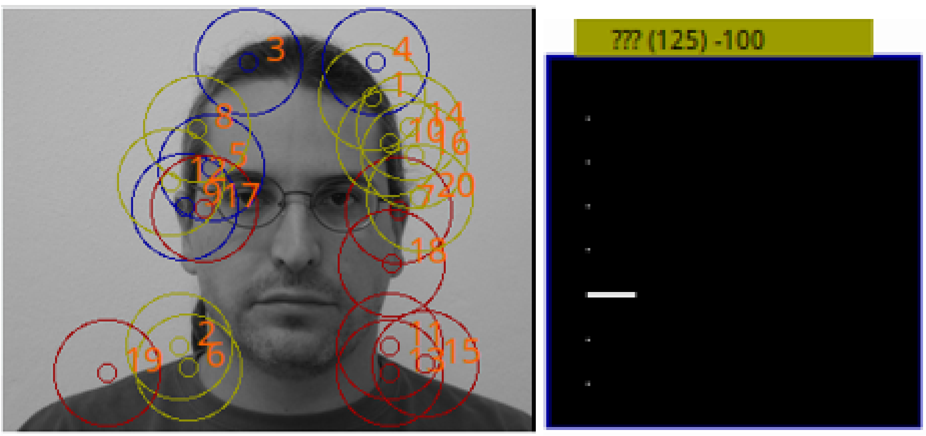
16.png



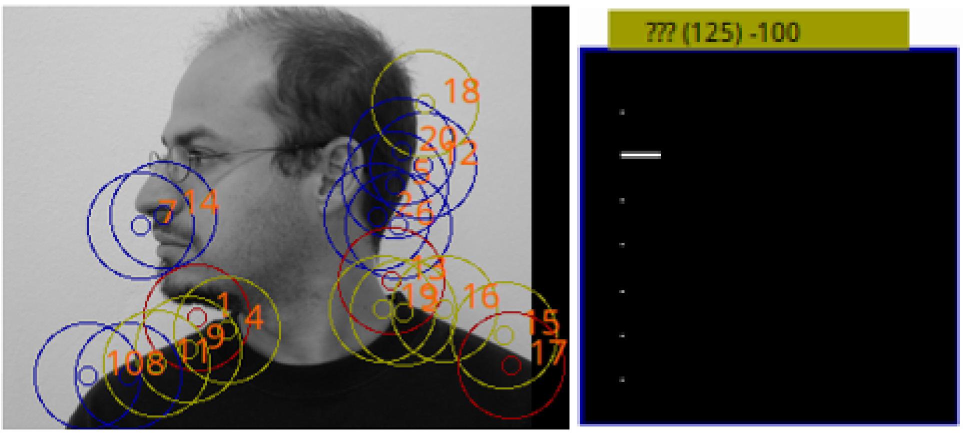
17.png



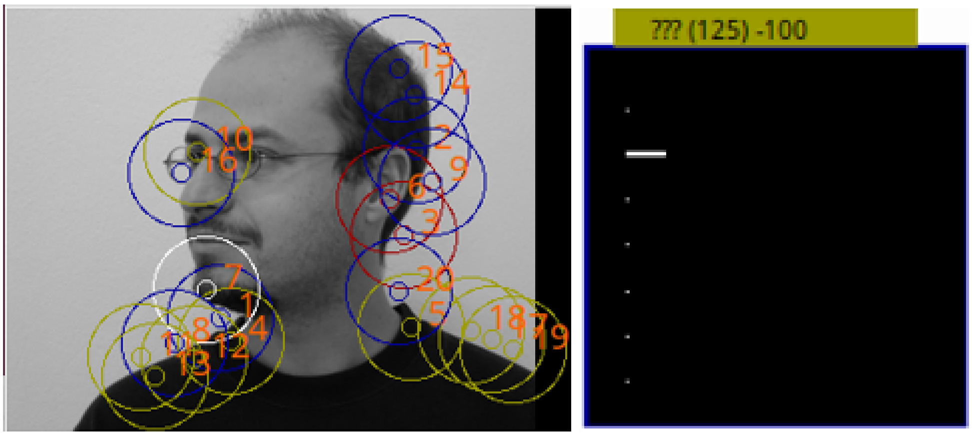
18.png



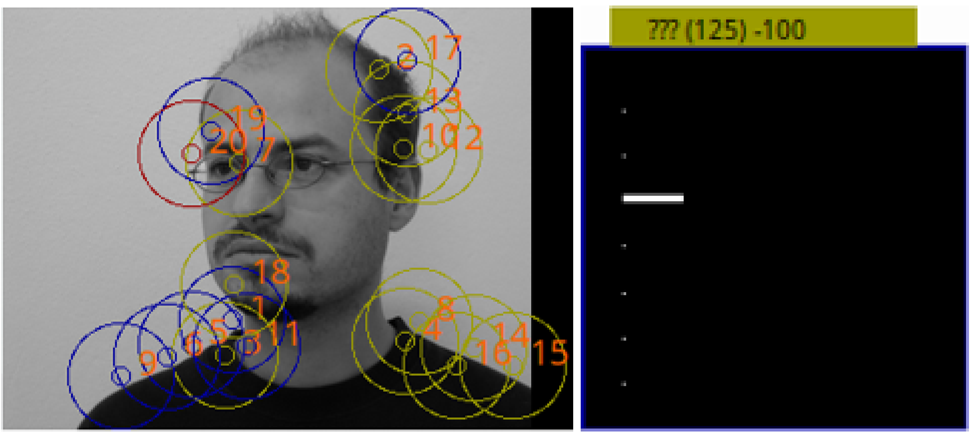
19.png



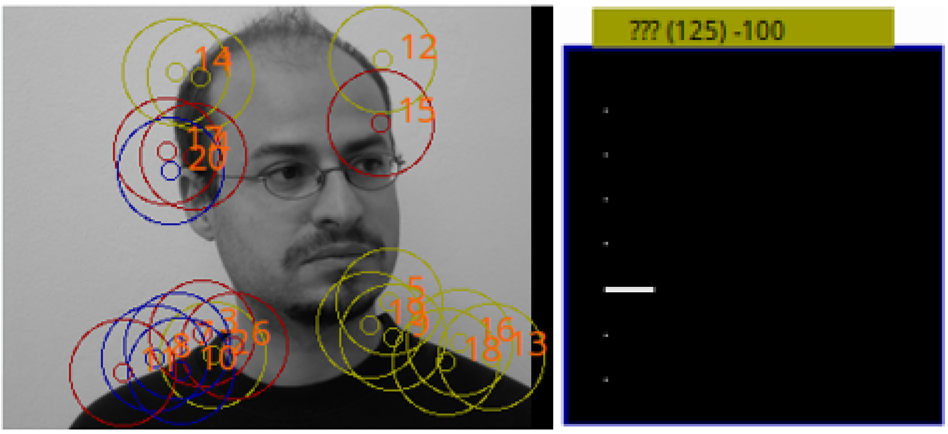
20.png



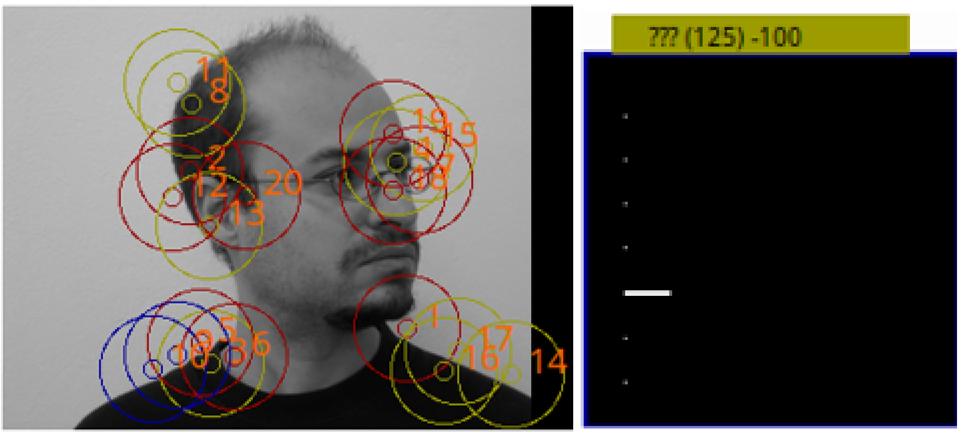
21.png



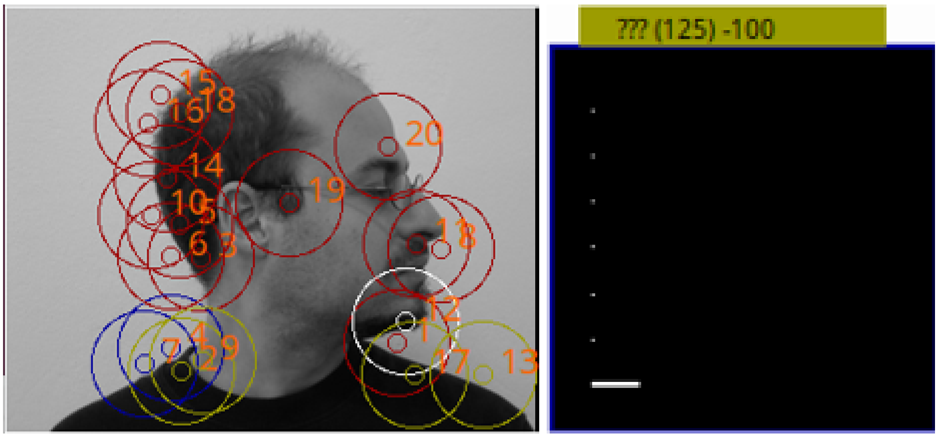
22.png



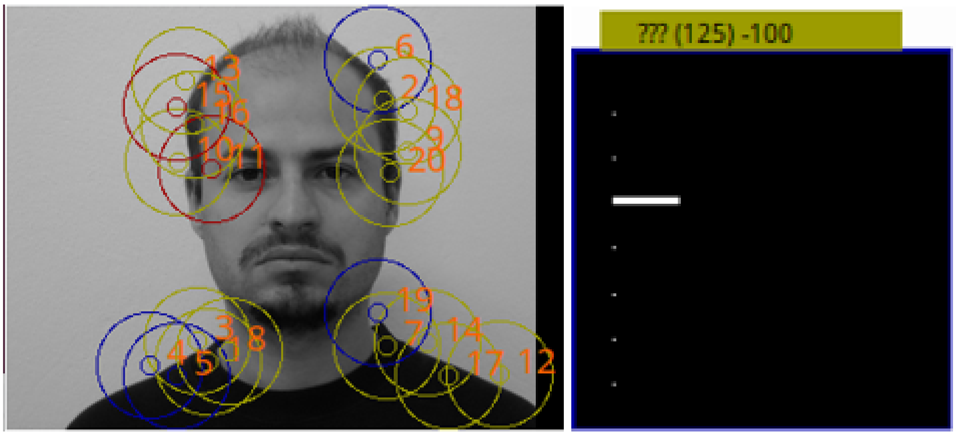
23.png



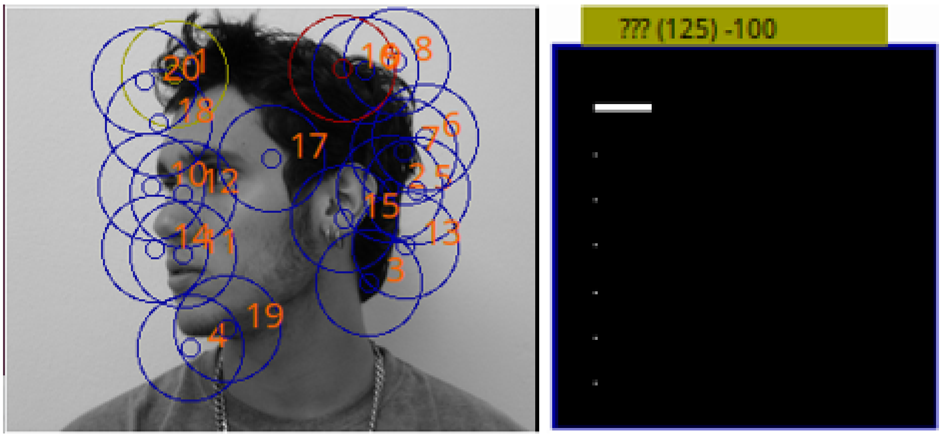
24.png



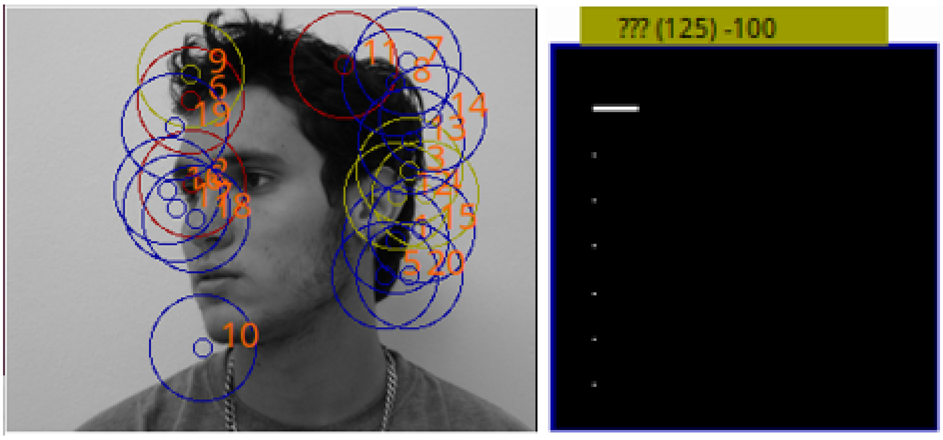
25.png



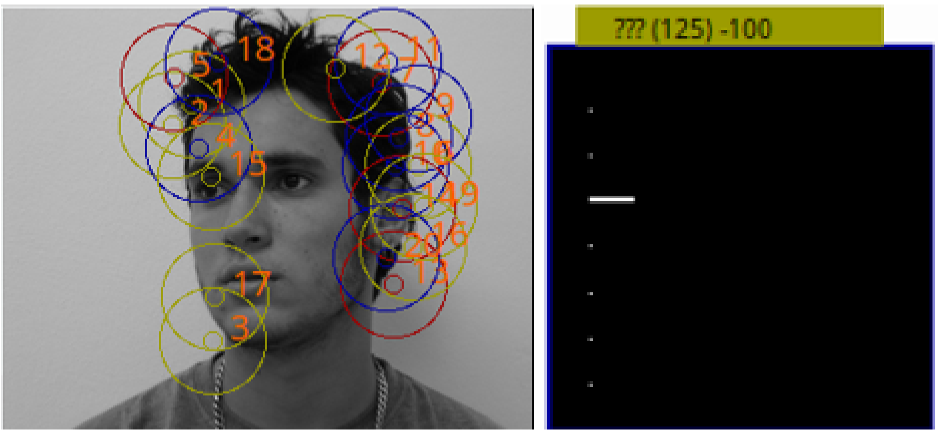
26.png



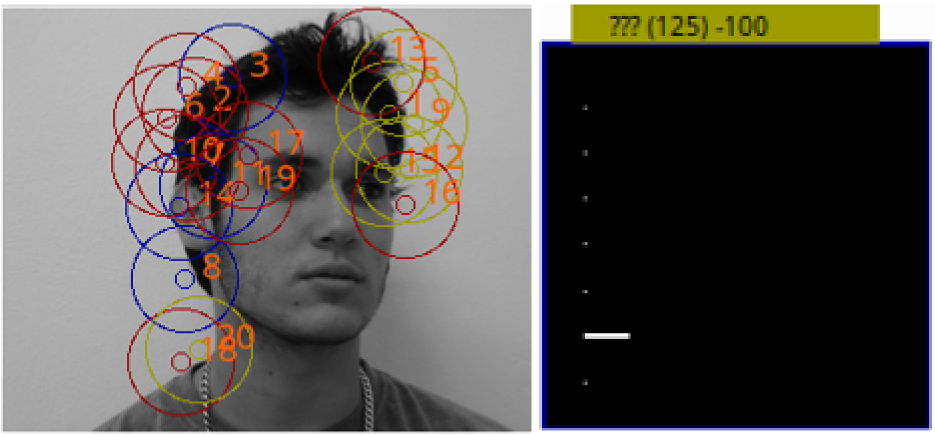
27.png



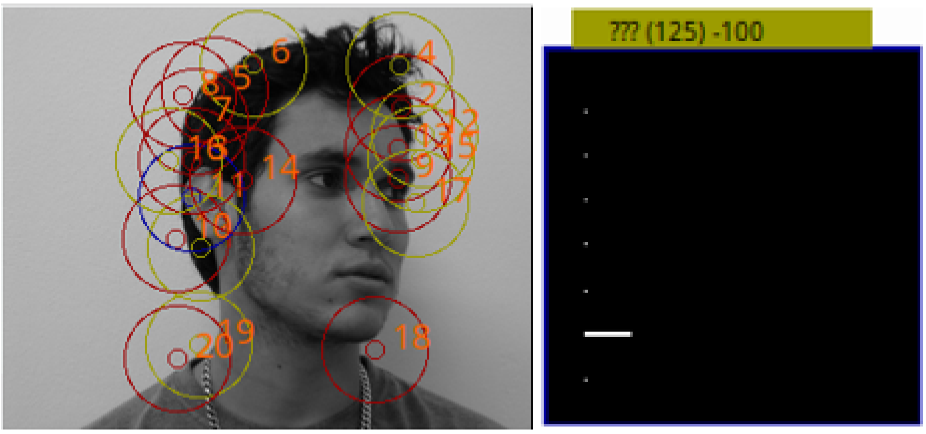
28.png



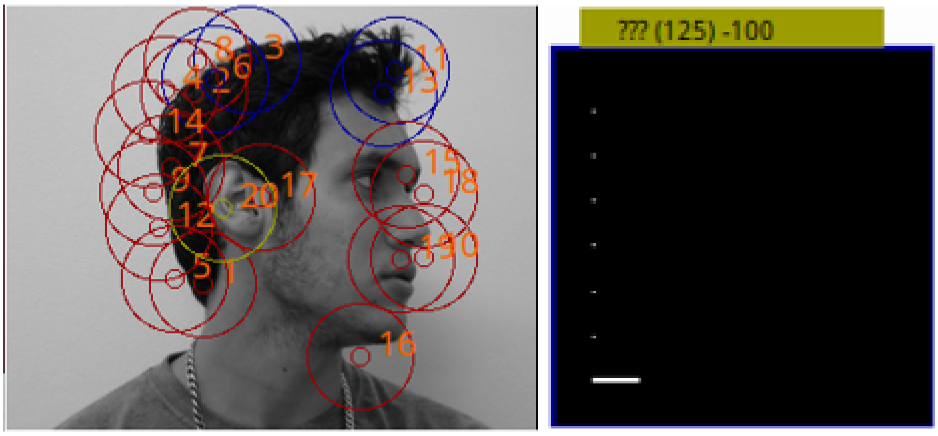
29.png



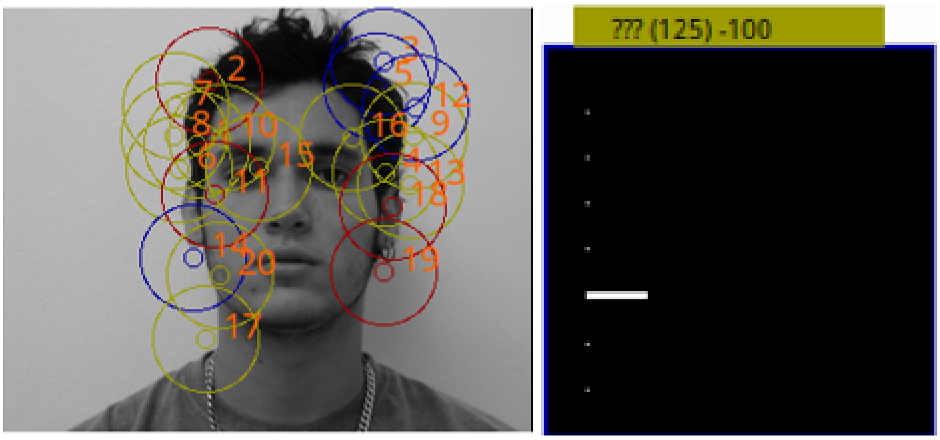
30.png



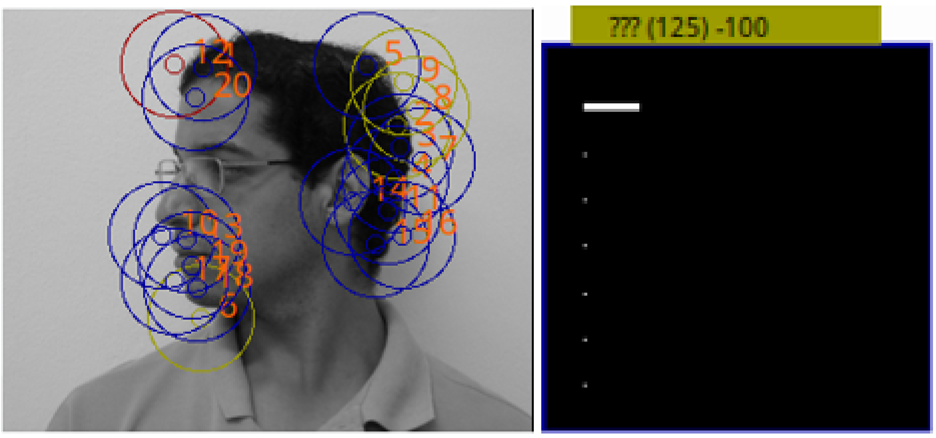
31.png



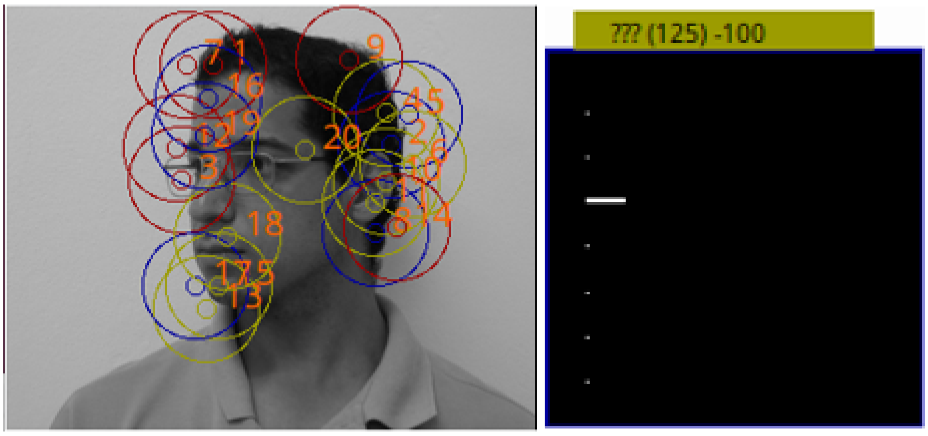
32.png



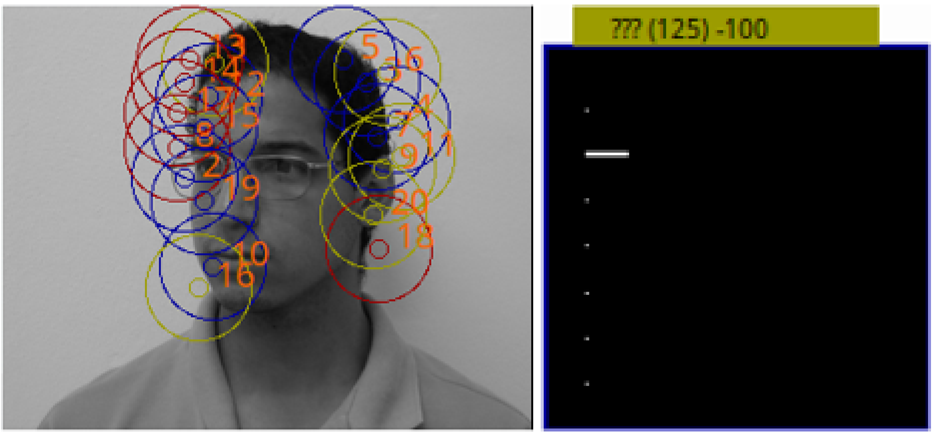
33.png



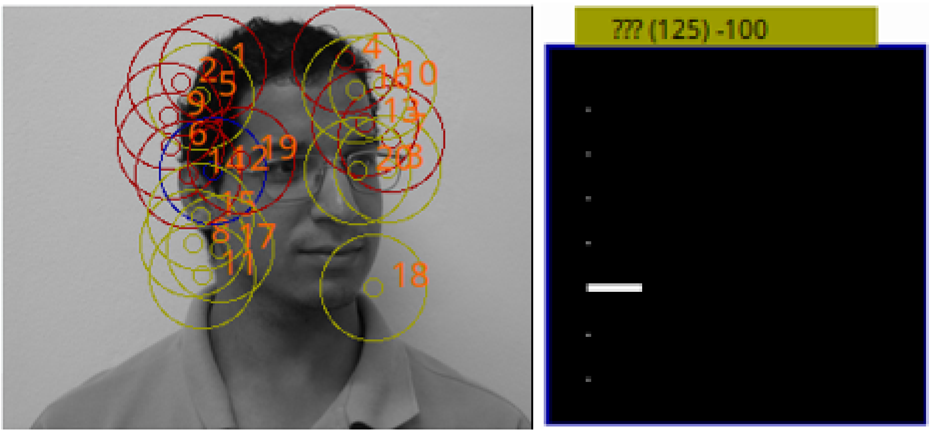
34.png



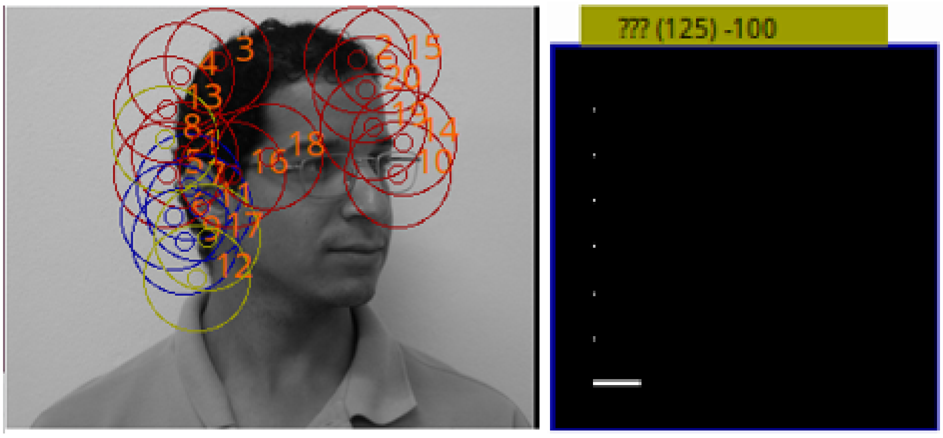
35.png



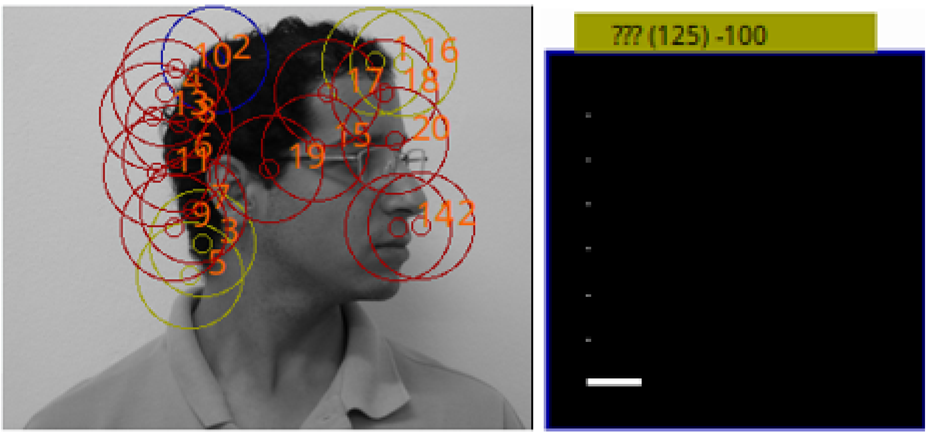
36.png



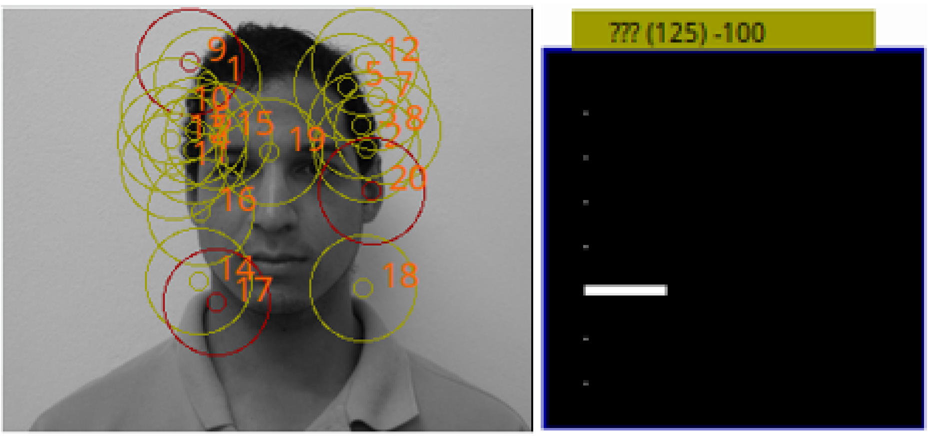
37.png



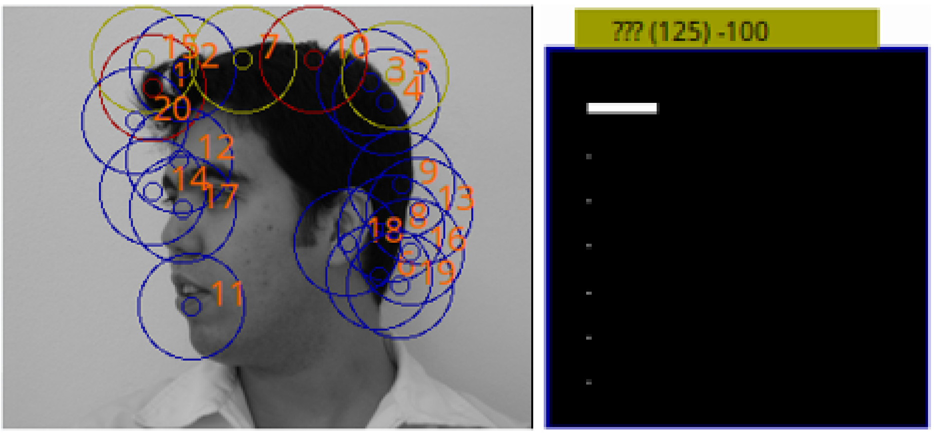
38.png



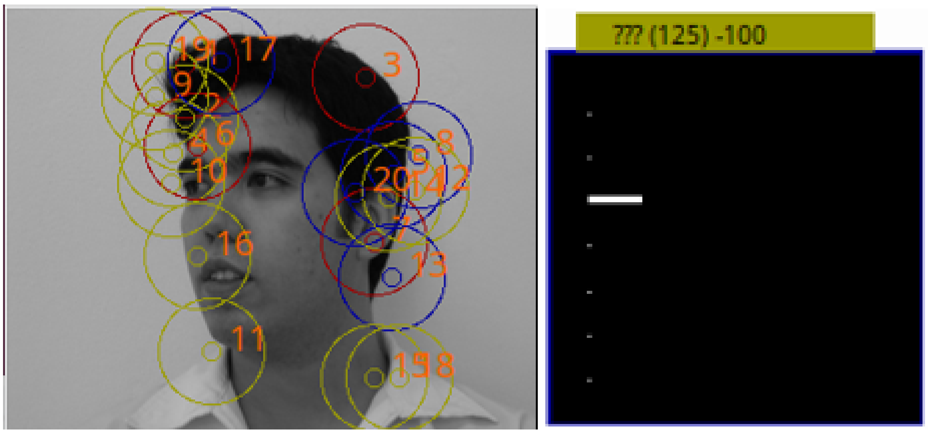
39.png



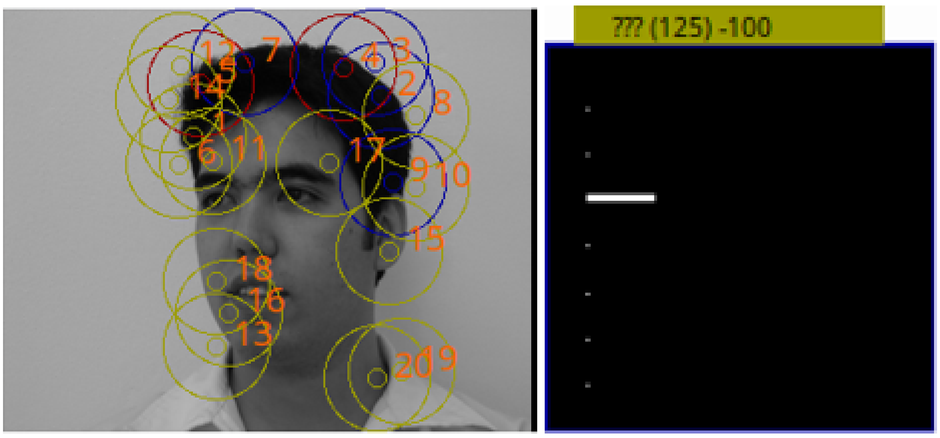
40.png



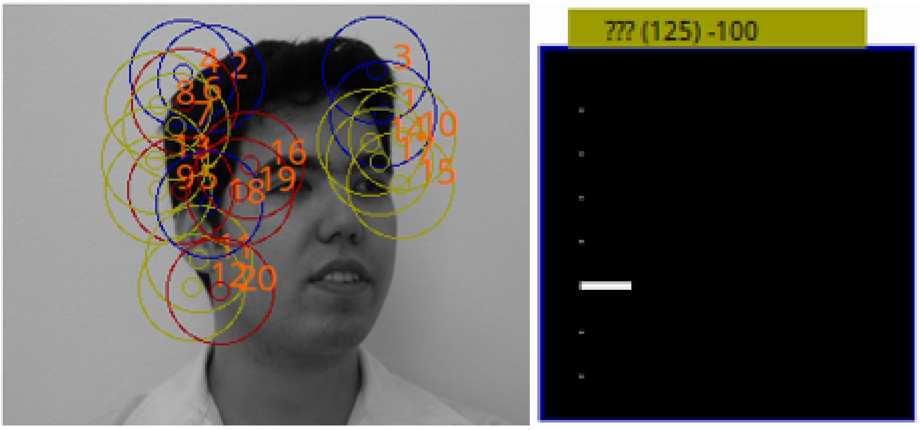
41.png



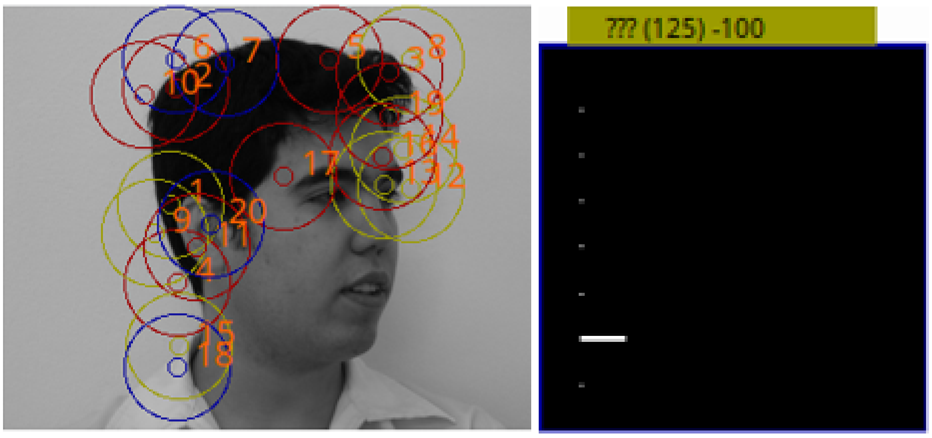
42.png



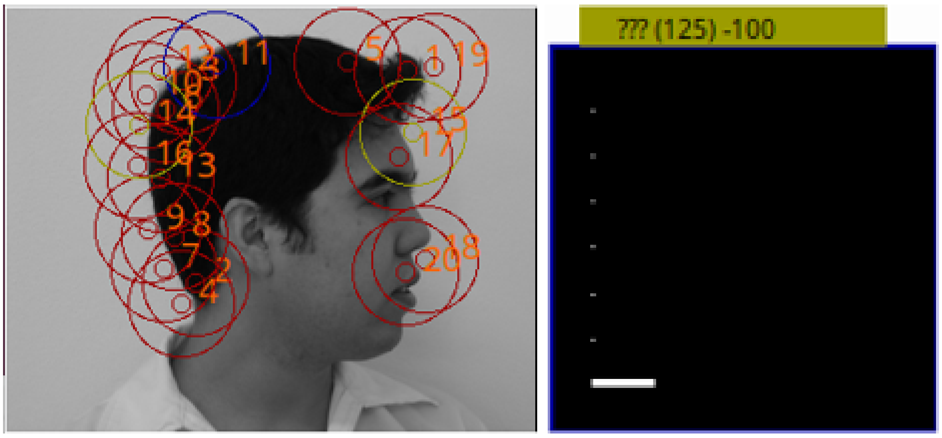
43.png



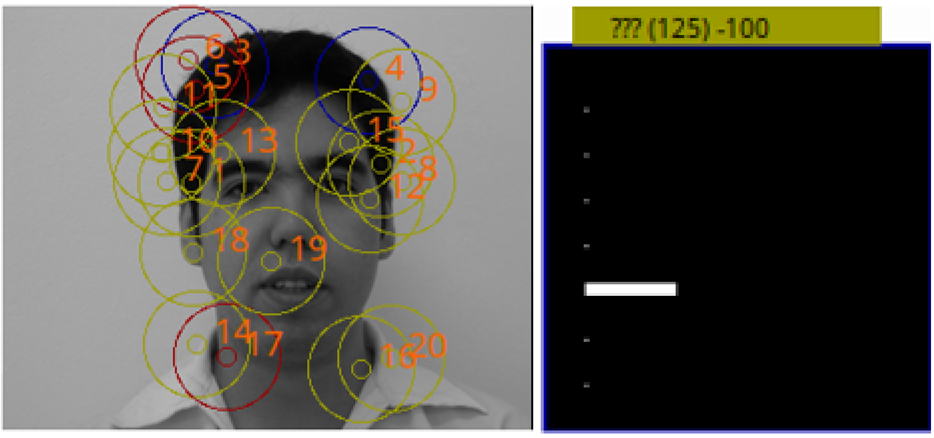
44.png



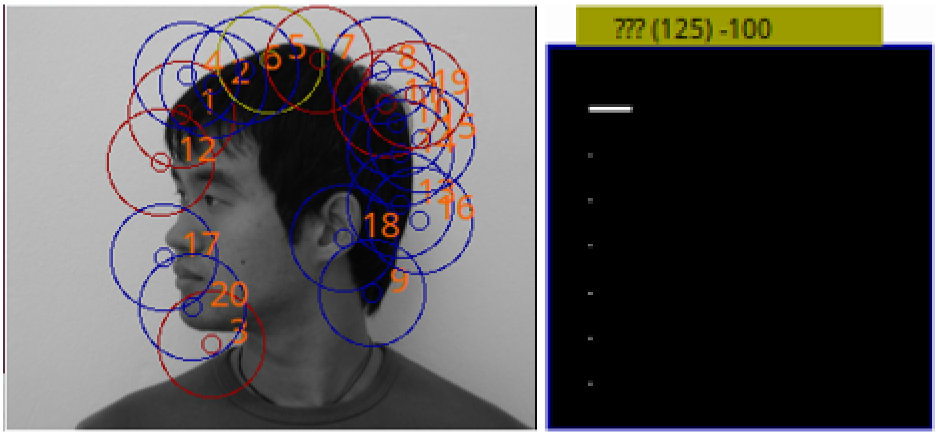
45.png



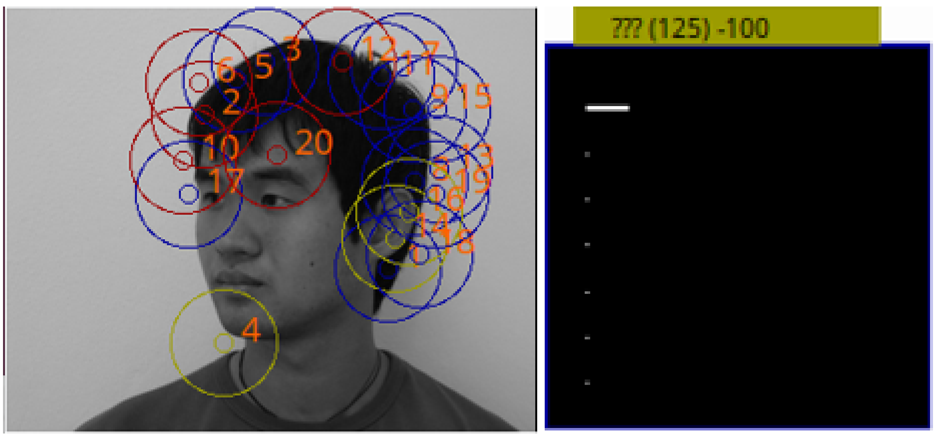
46.png



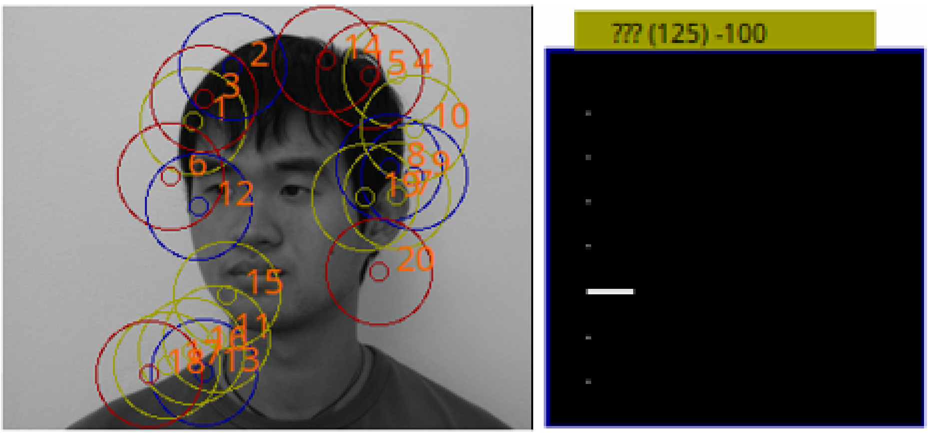
47.png



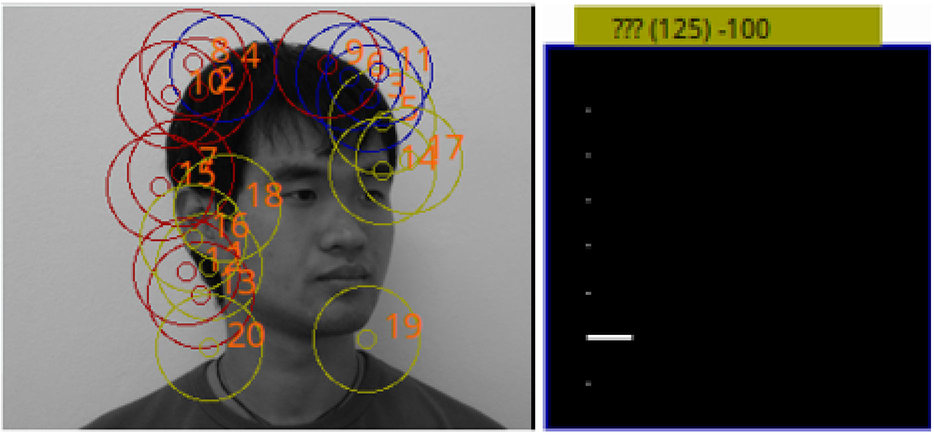
48.png



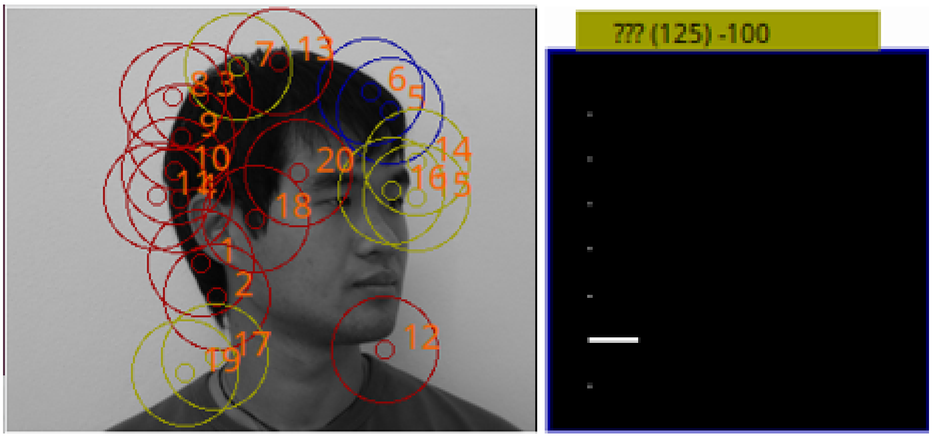
49.png



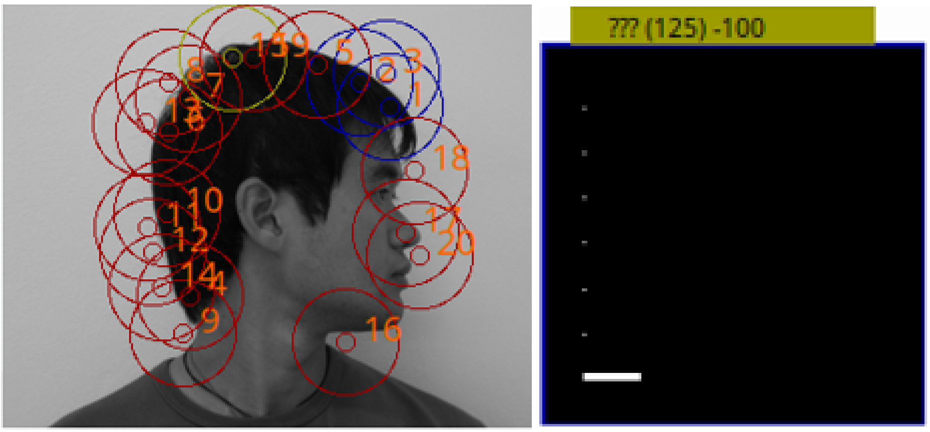
50.png



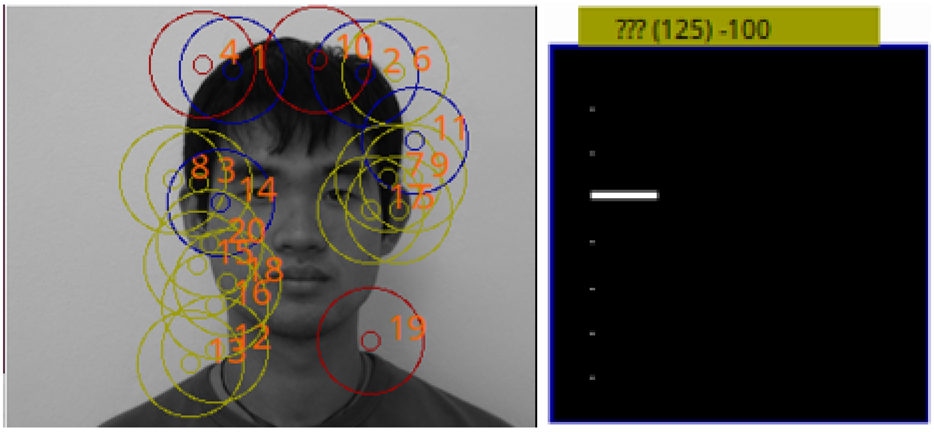
51.png



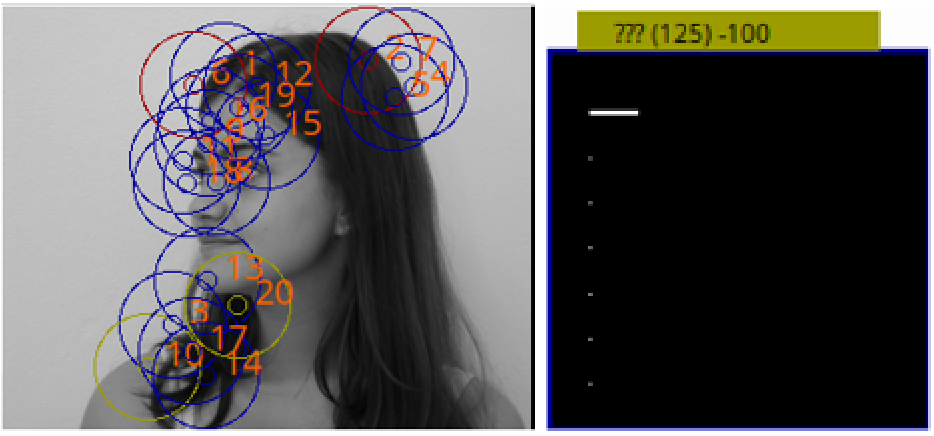
52.png



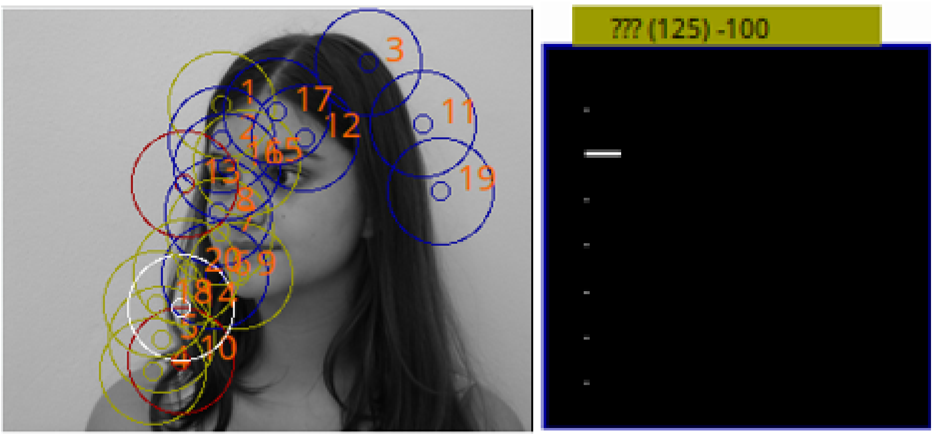
53.png



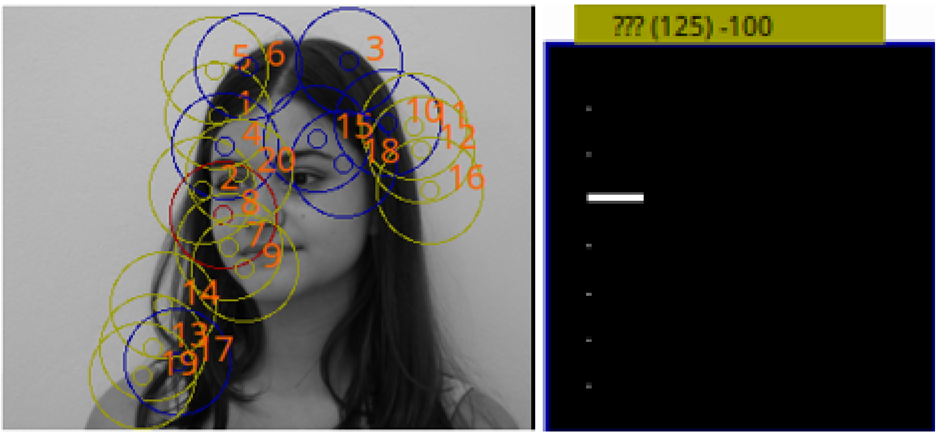
54.png



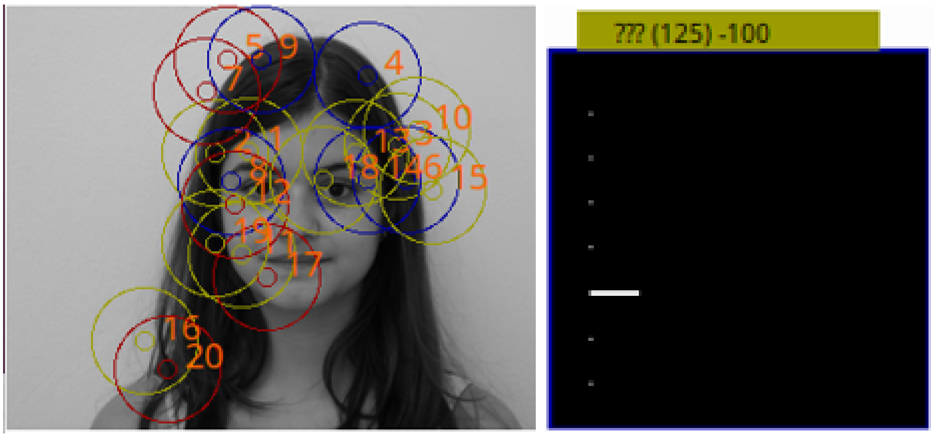
55.png



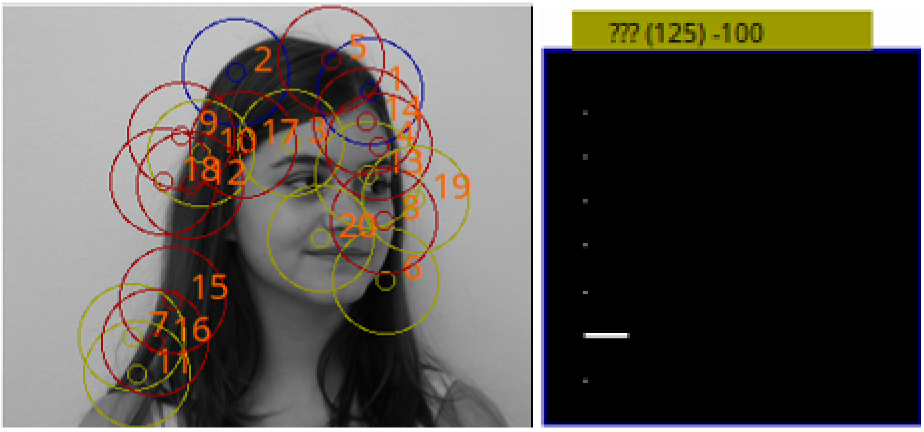
56.png



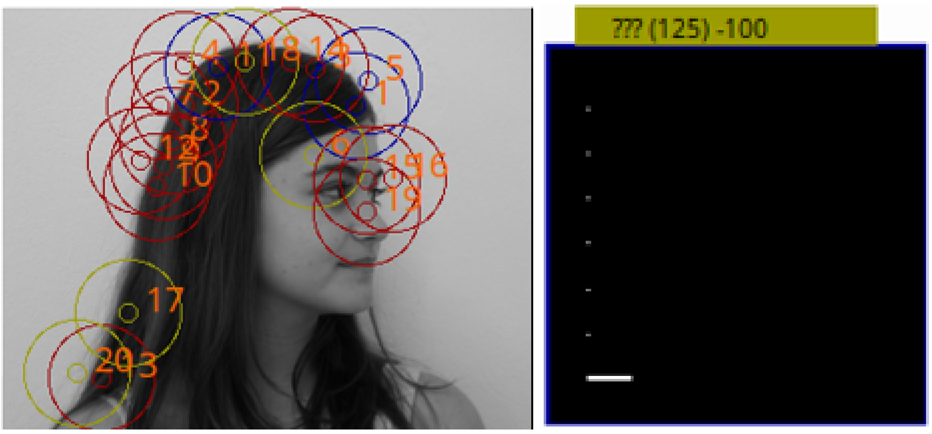
57.png



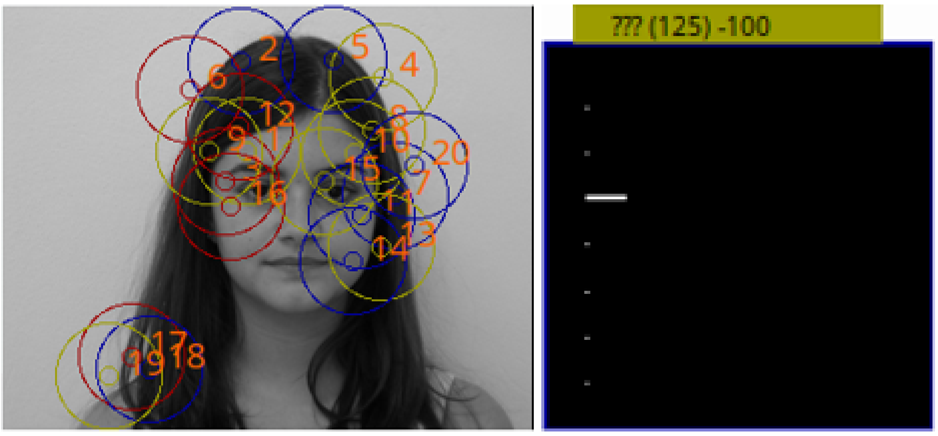
58.png



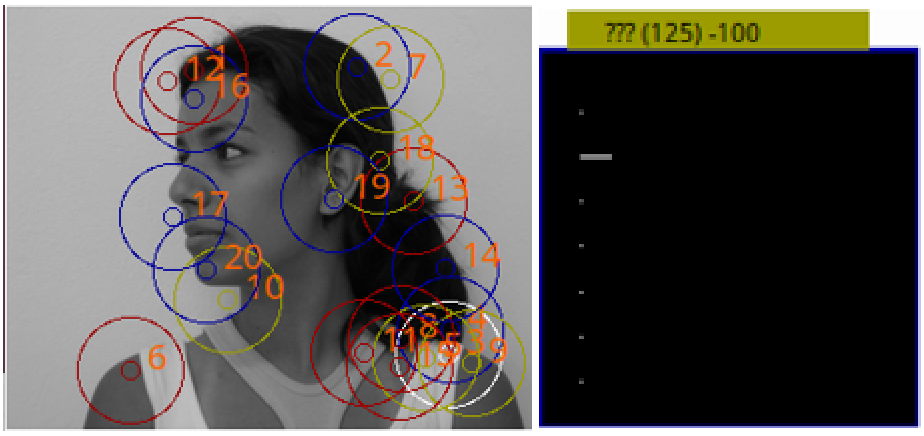
59.png



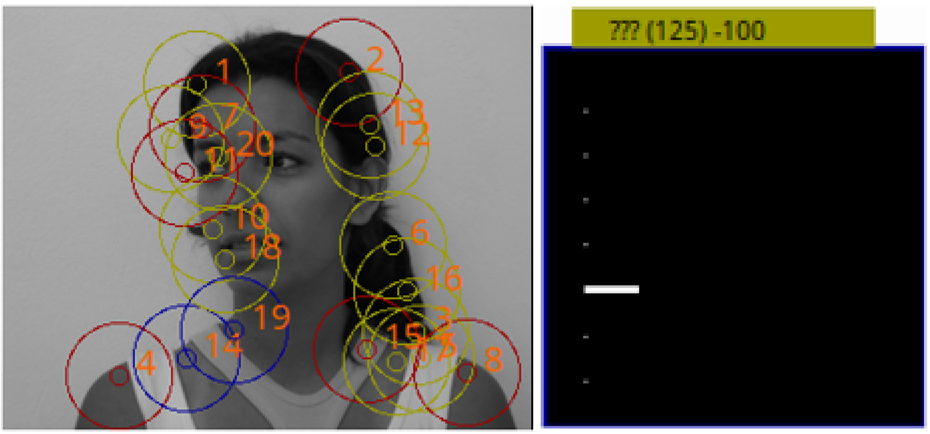
60.png



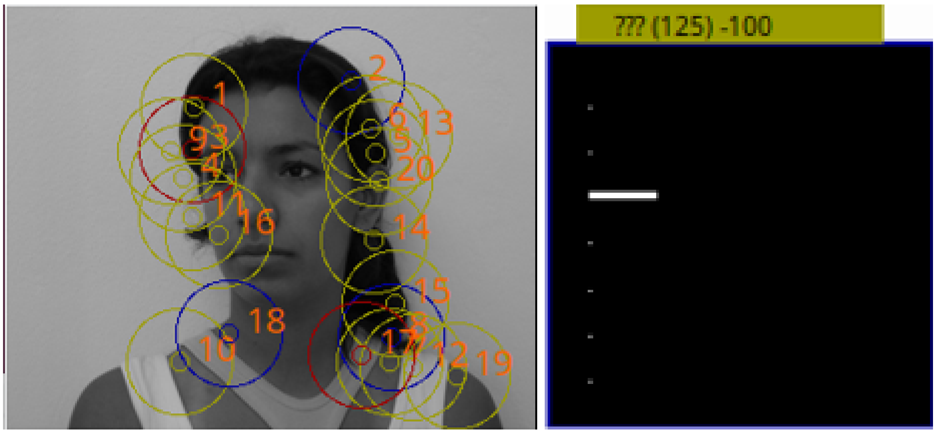
61.png



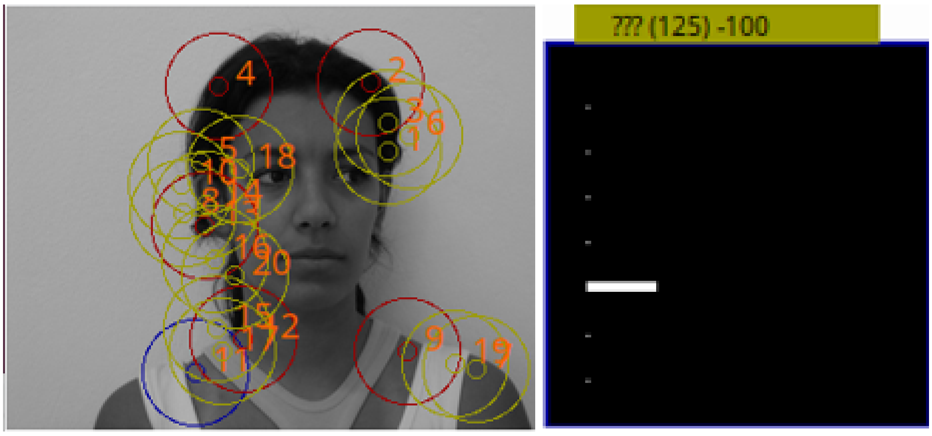
62.png



63.png

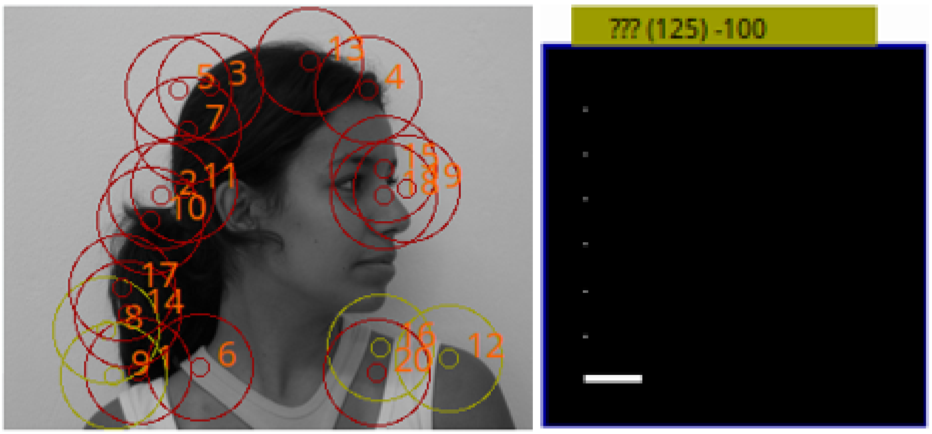


64.png

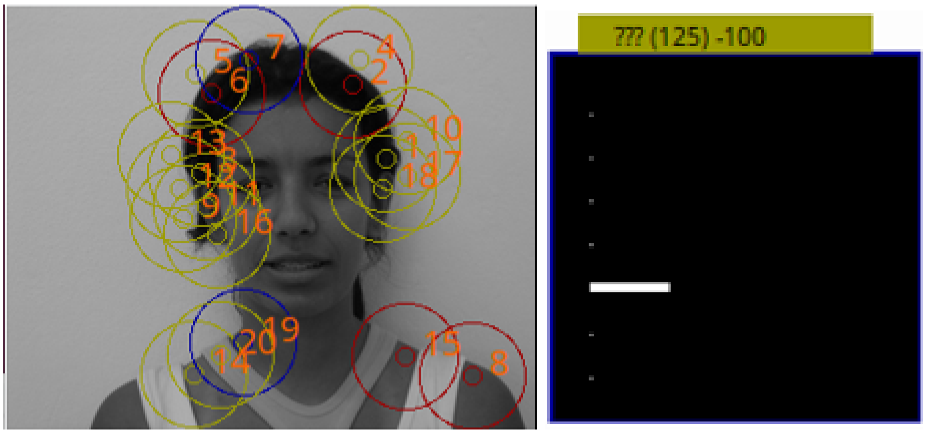


65.png

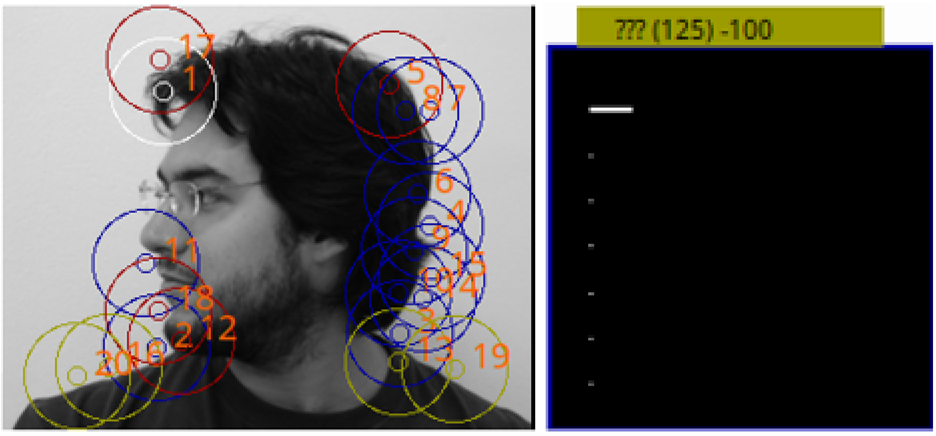
66.png



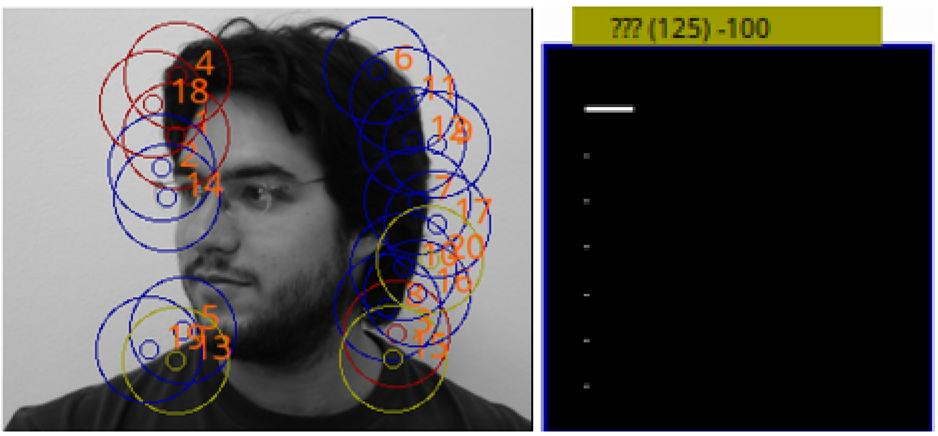
67.png



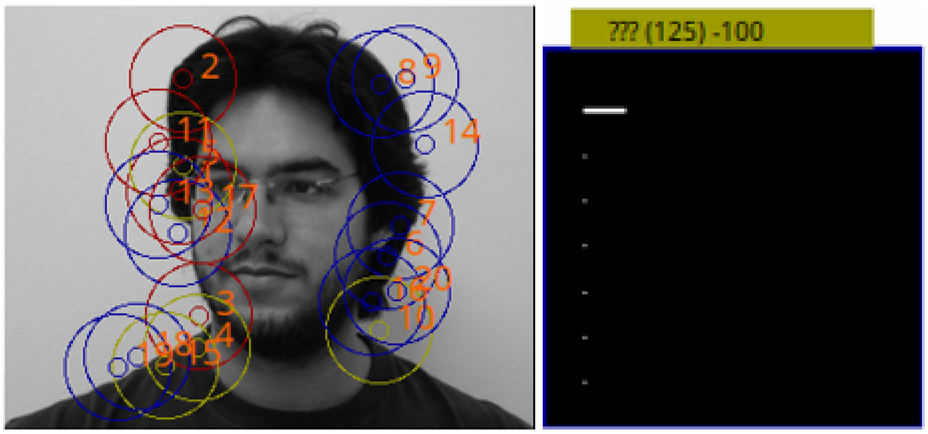
68.png



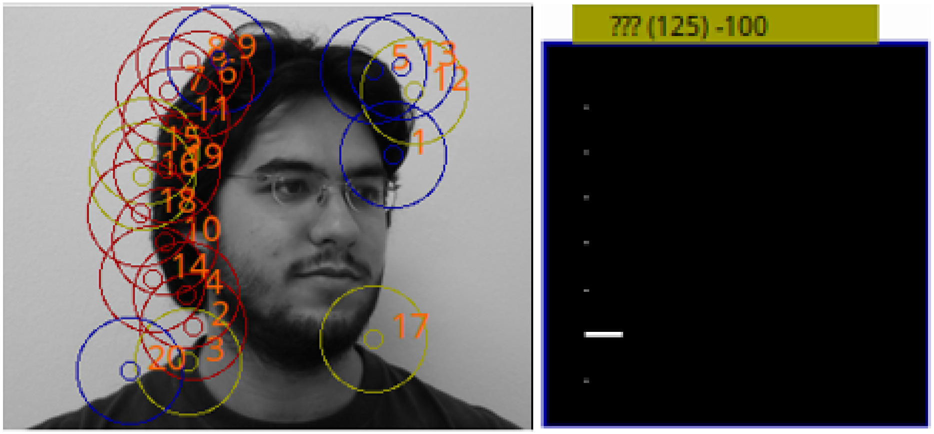
69.png



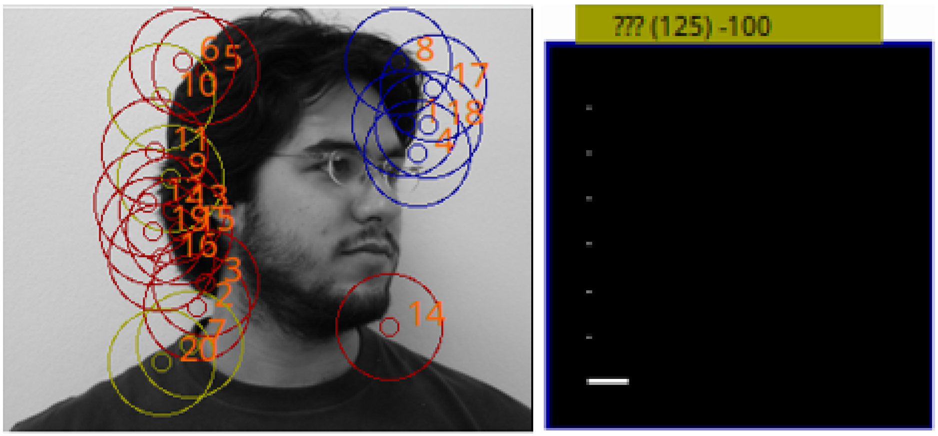
70.png



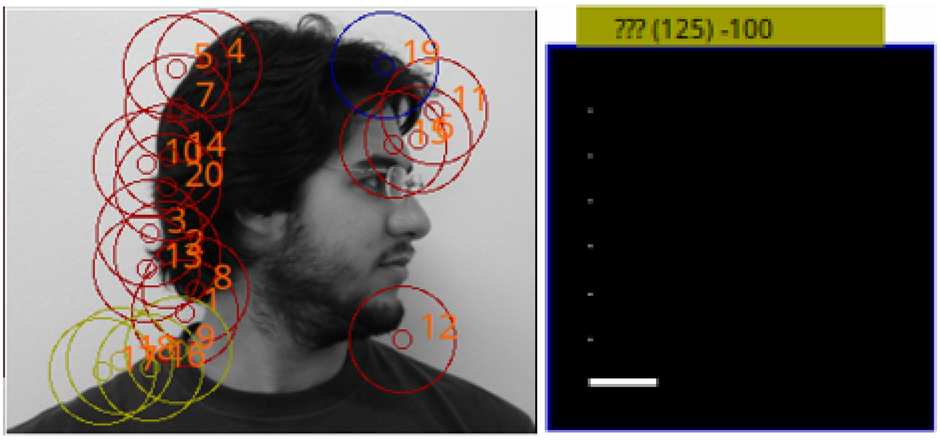
71.png



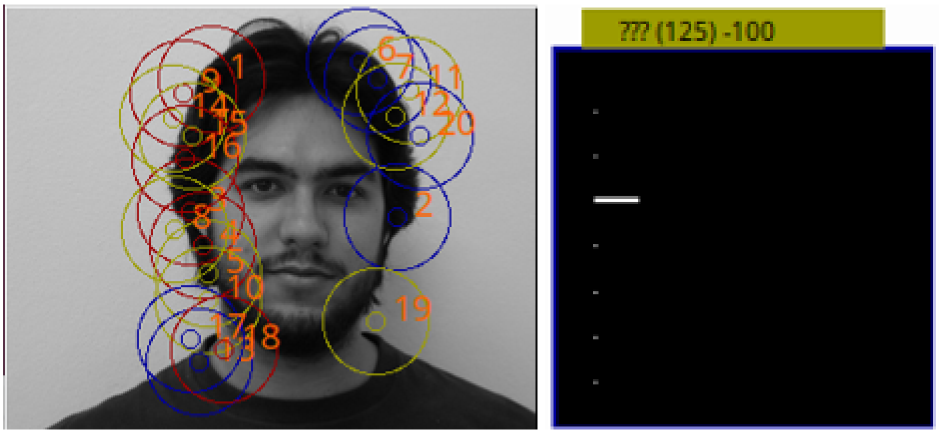
72.png



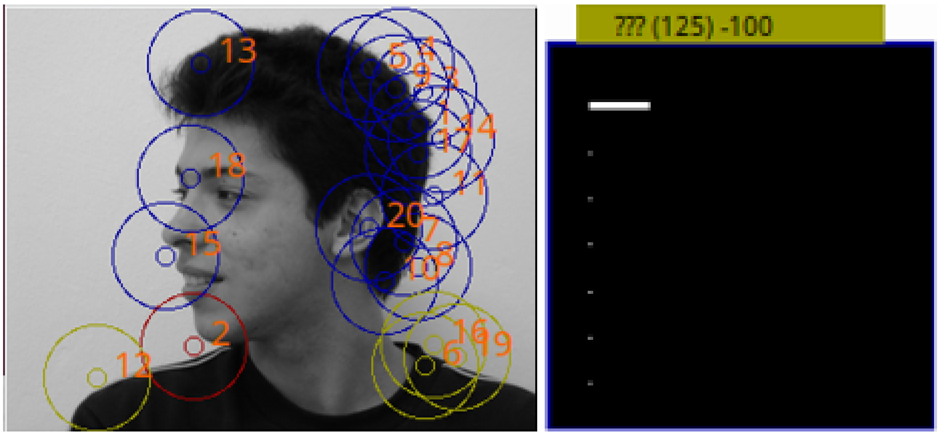
73.png



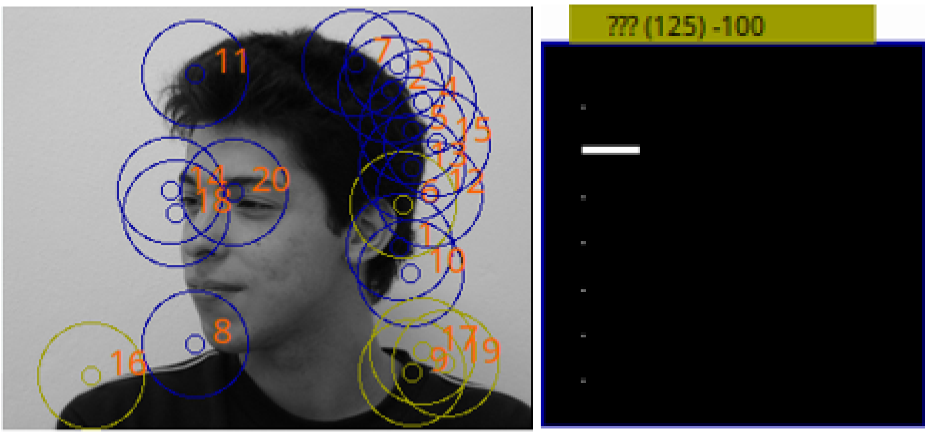
74.png



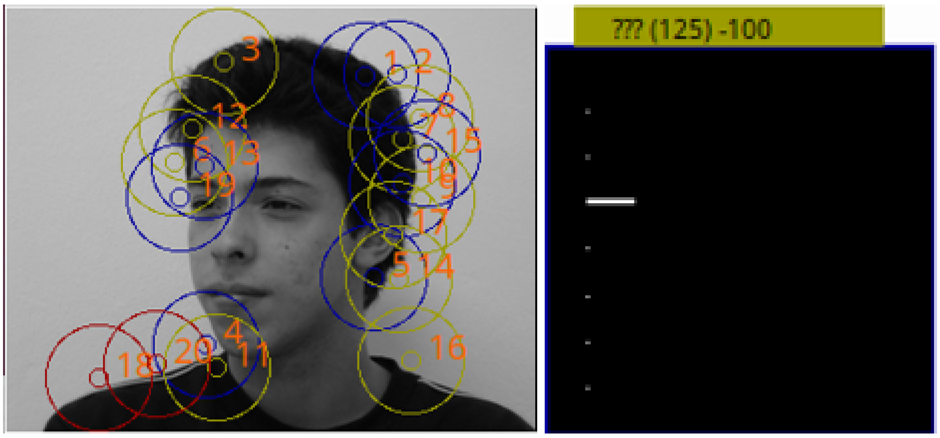
75.png



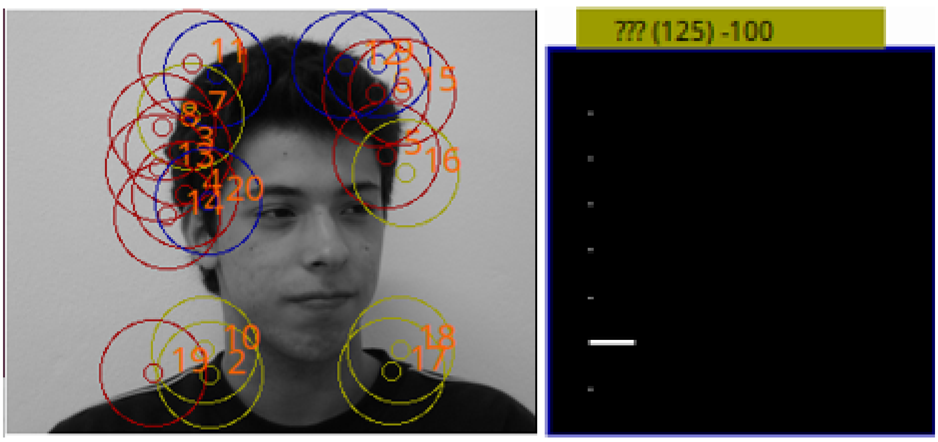
76.png



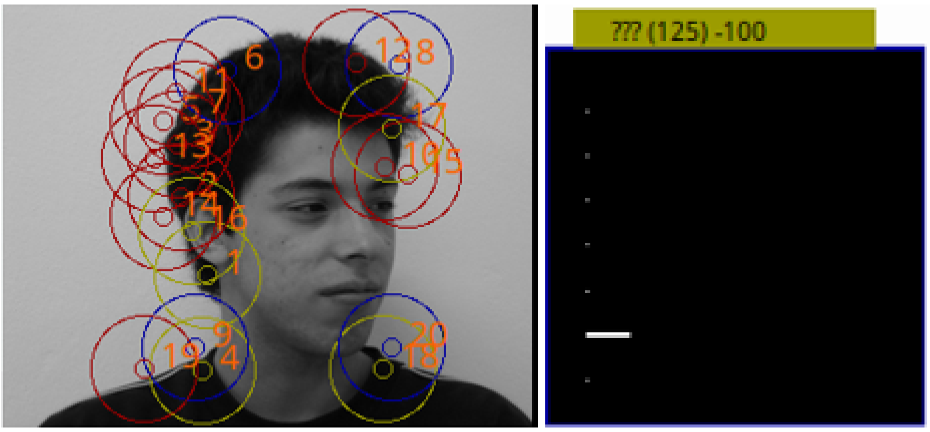
77.png



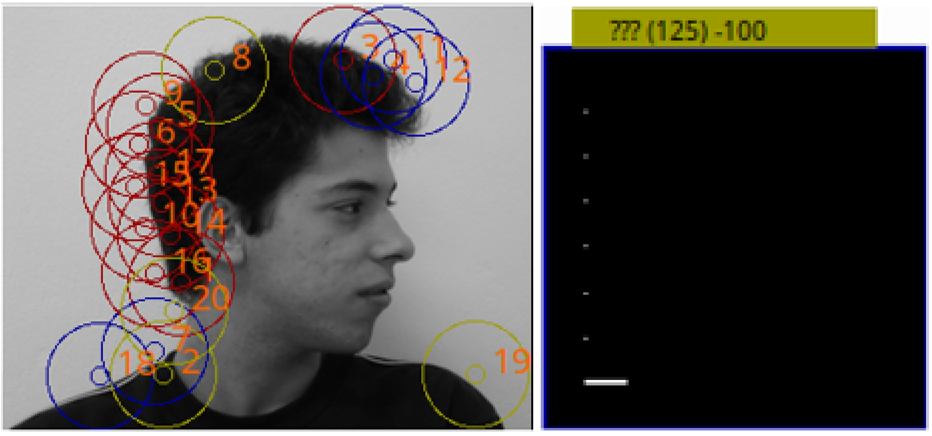
78.png



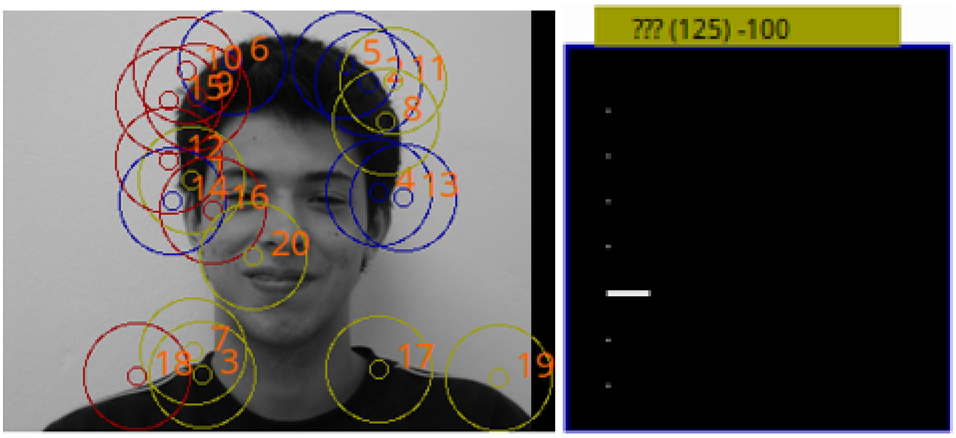
79.png



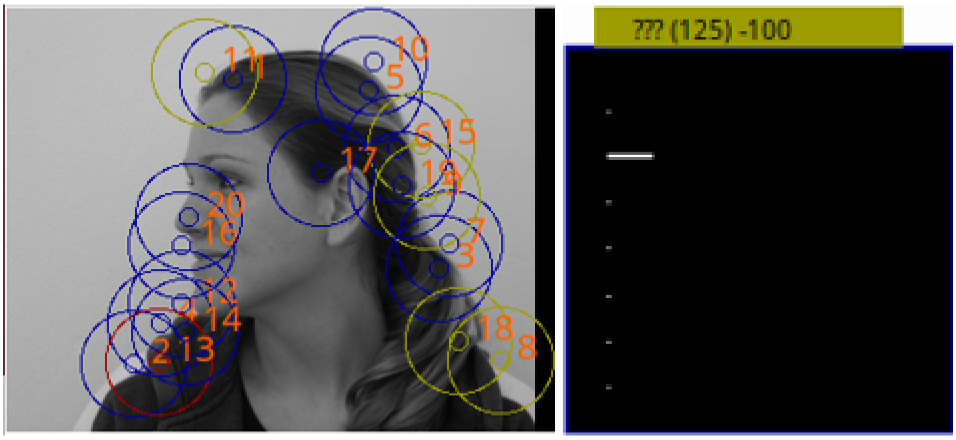
80.png



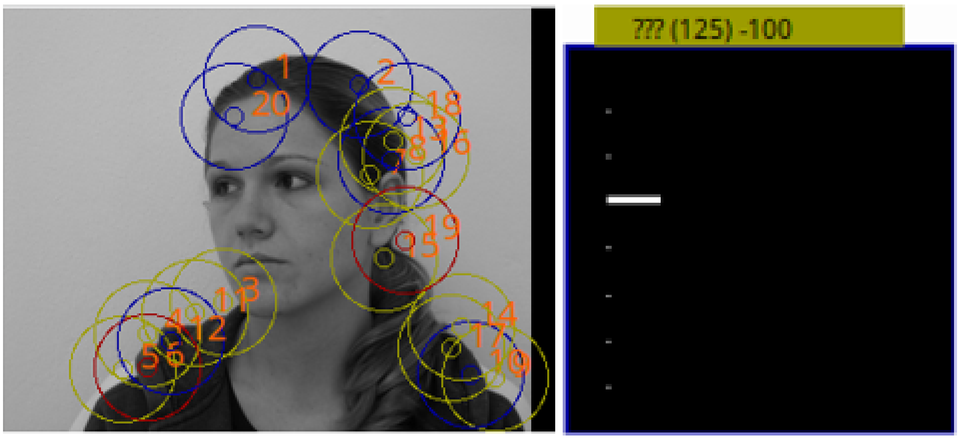
81.png



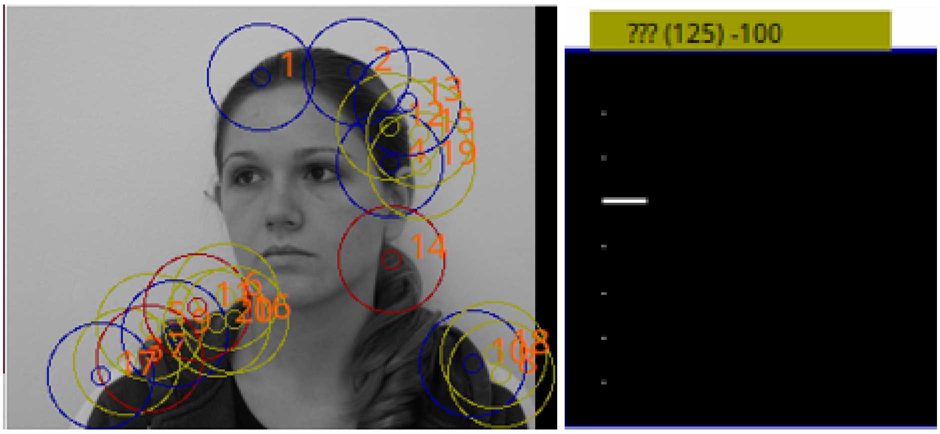
82.png



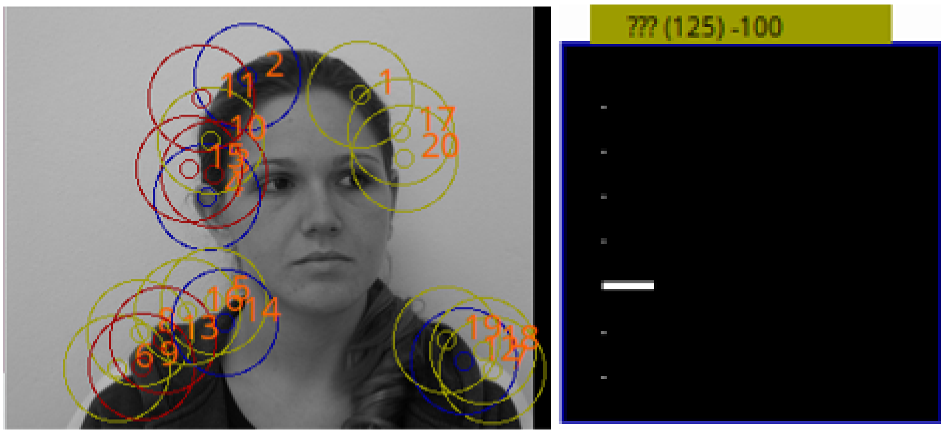
83.png



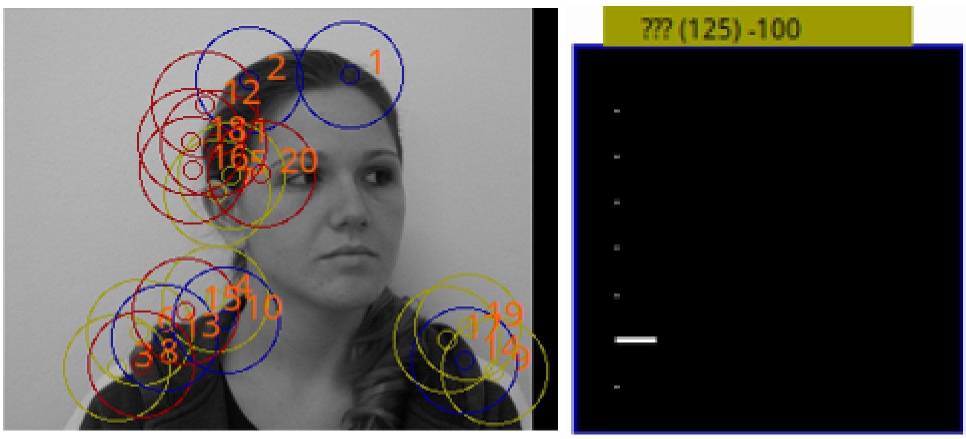
84.png



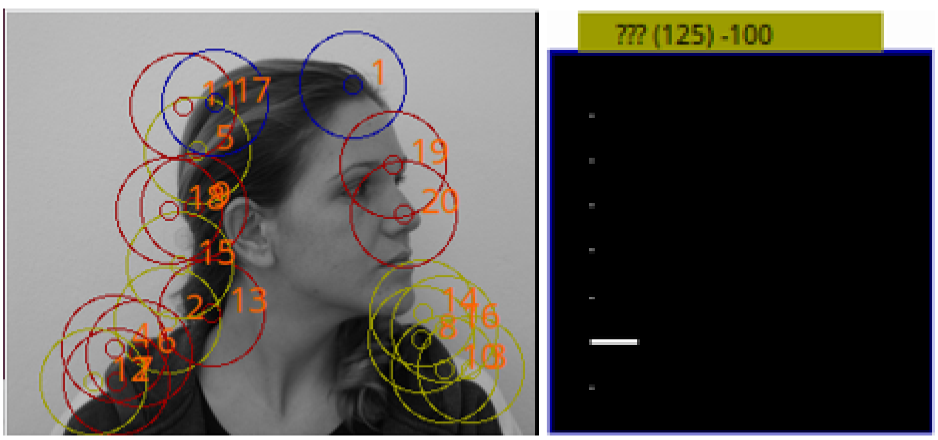
85.png



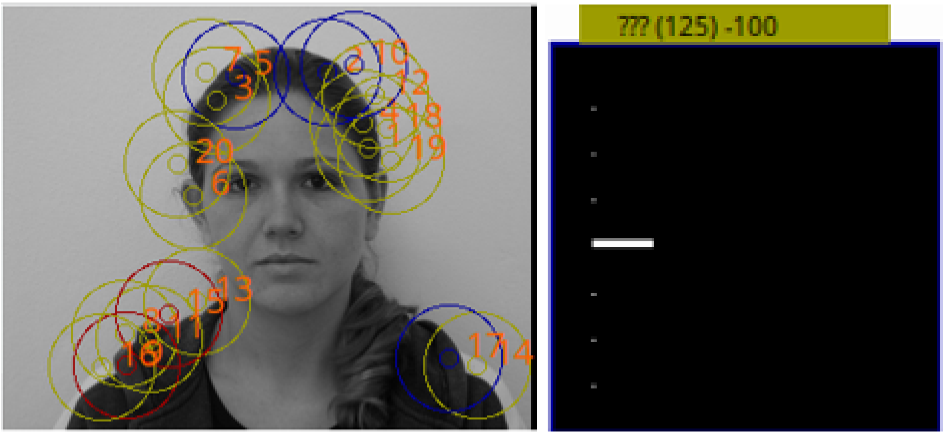
86.png



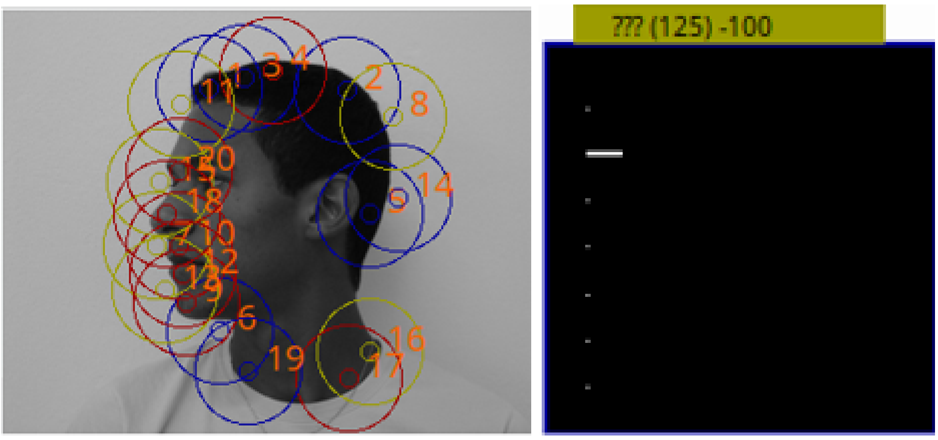
87.png



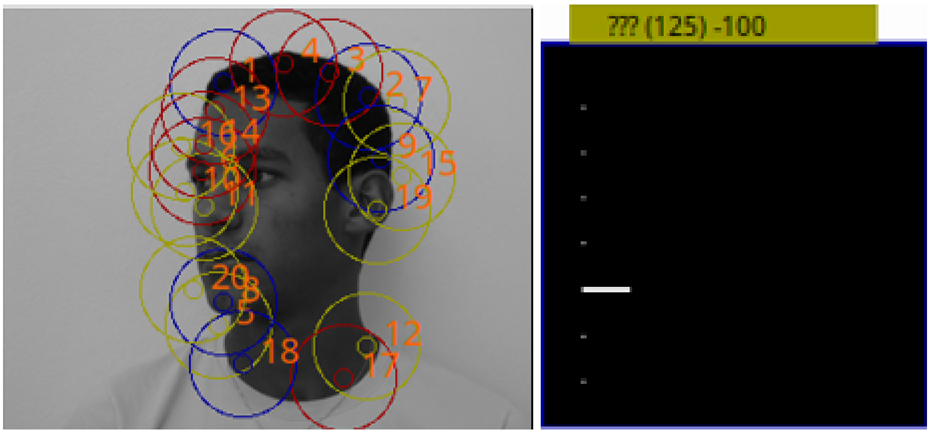
88.png



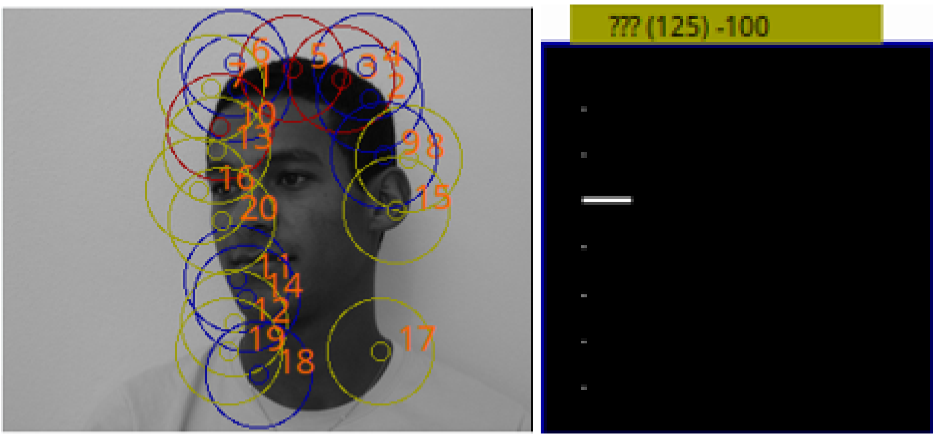
89.png



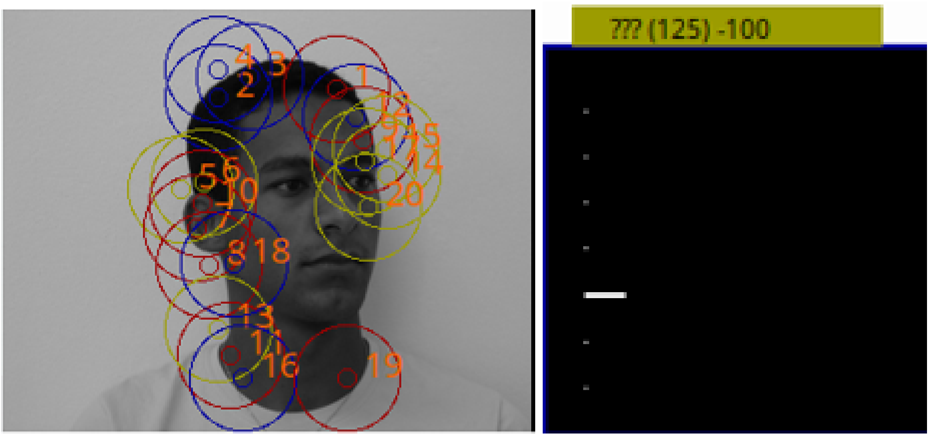
90.png



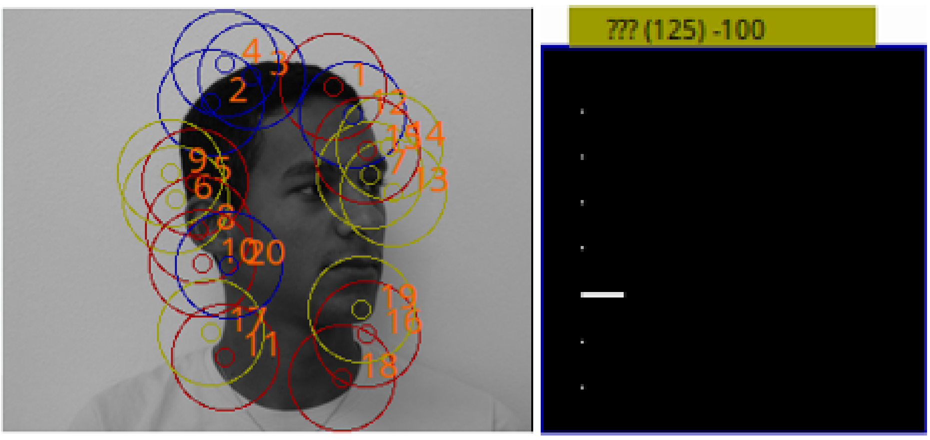
91.png



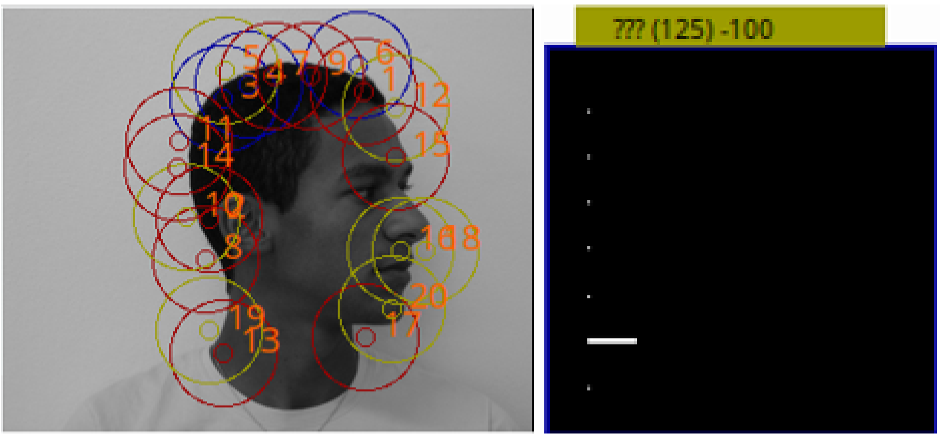
92.png



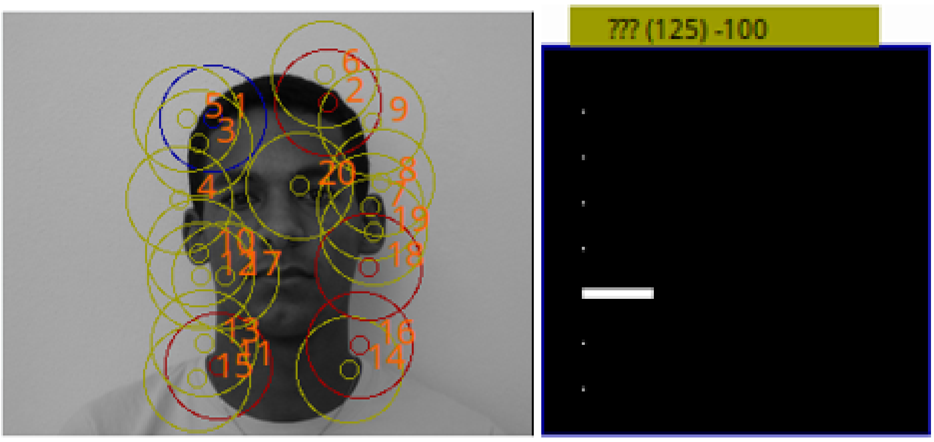
93.png



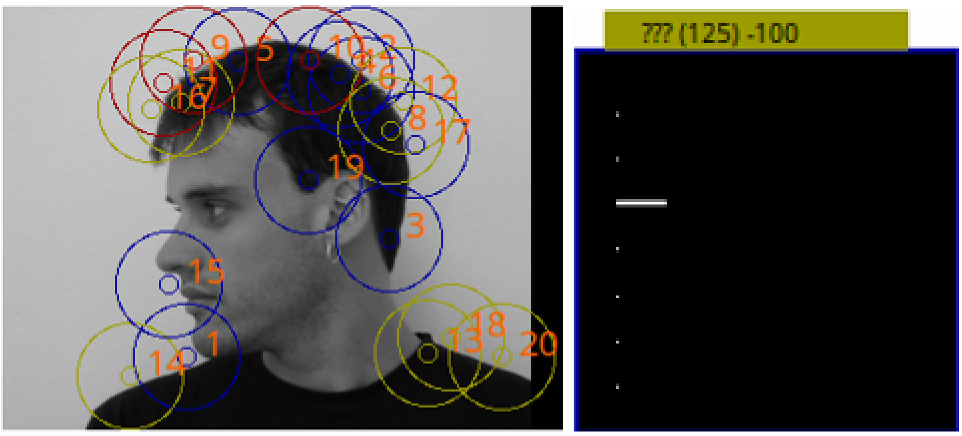
94.png



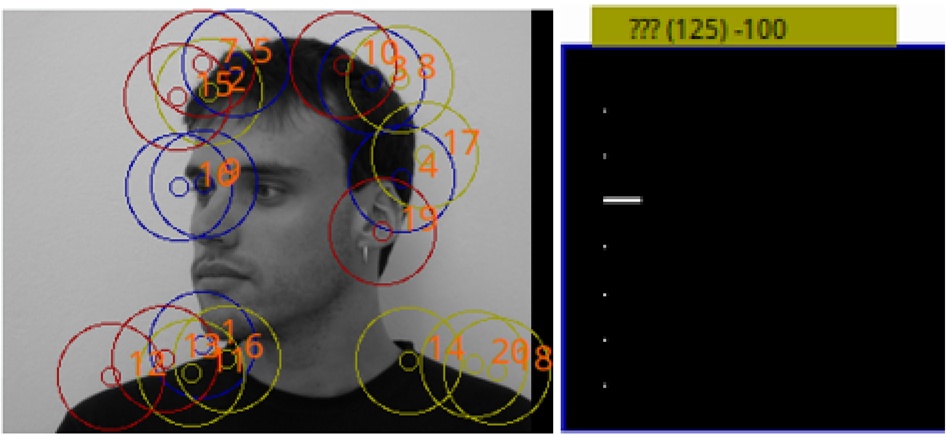
95.png



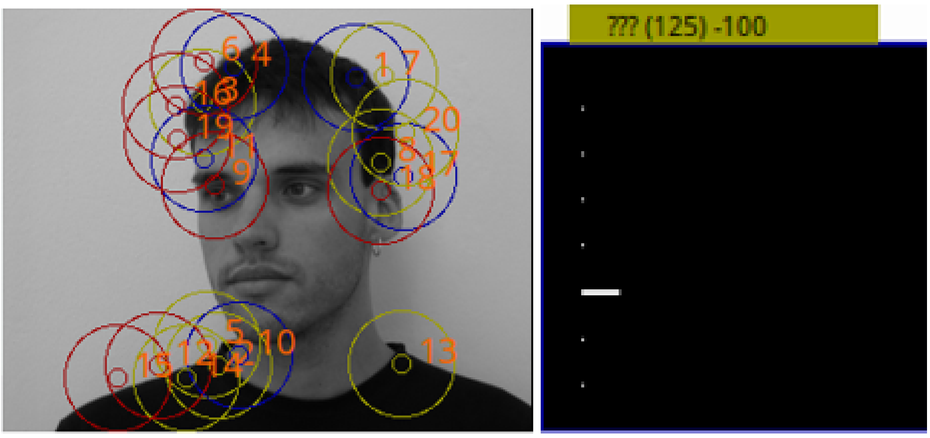
96.png



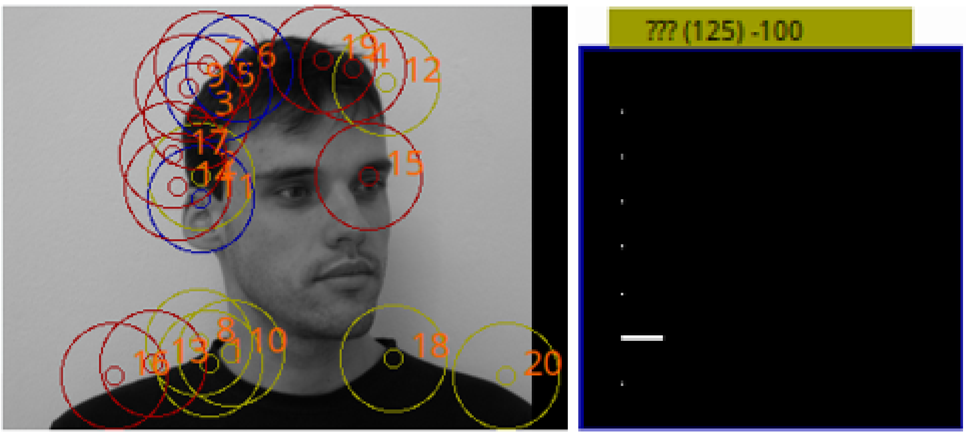
97.png



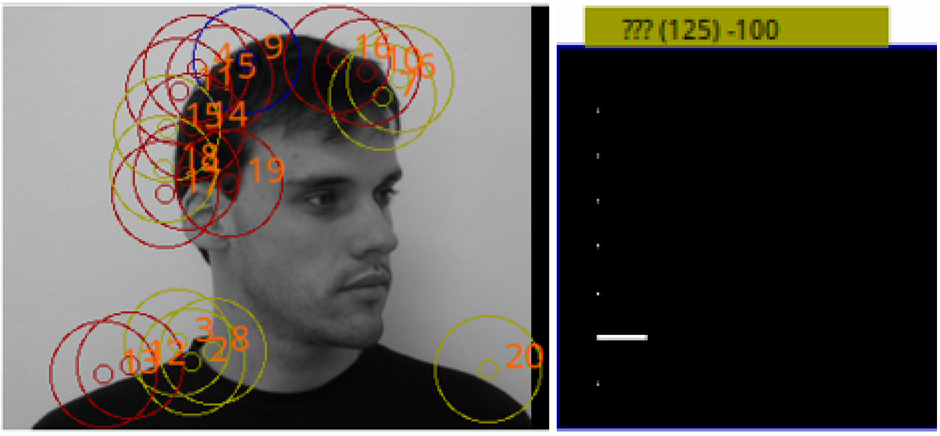
98.png



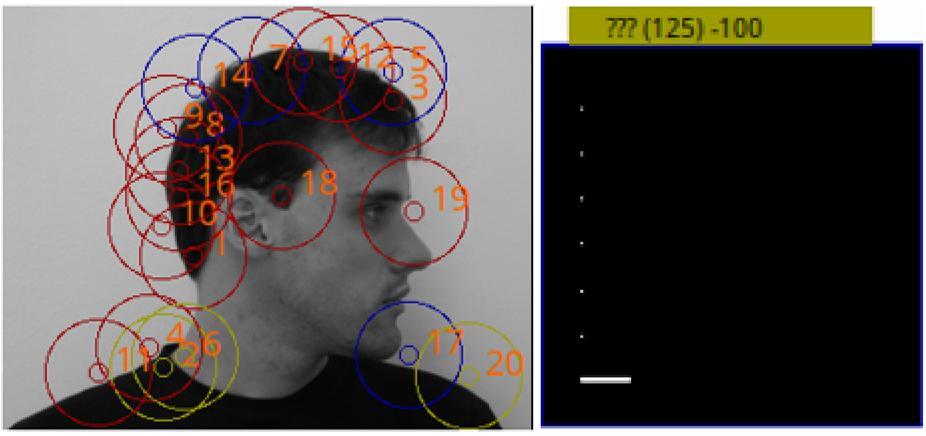
99.png



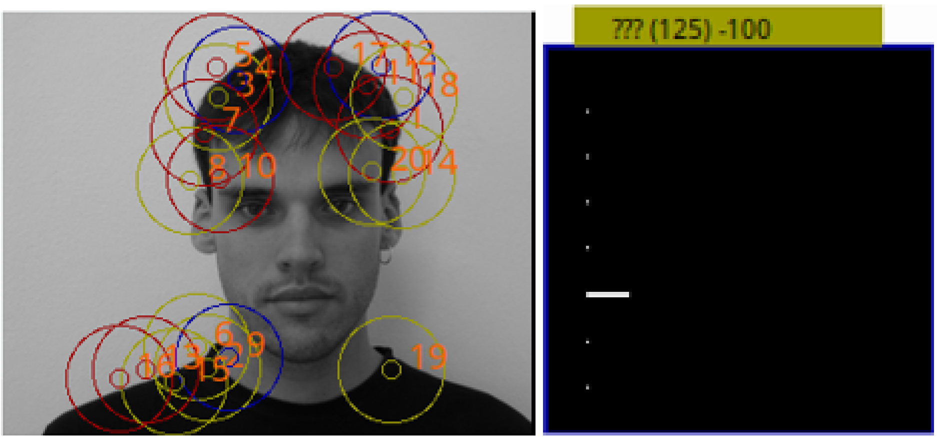
100.png



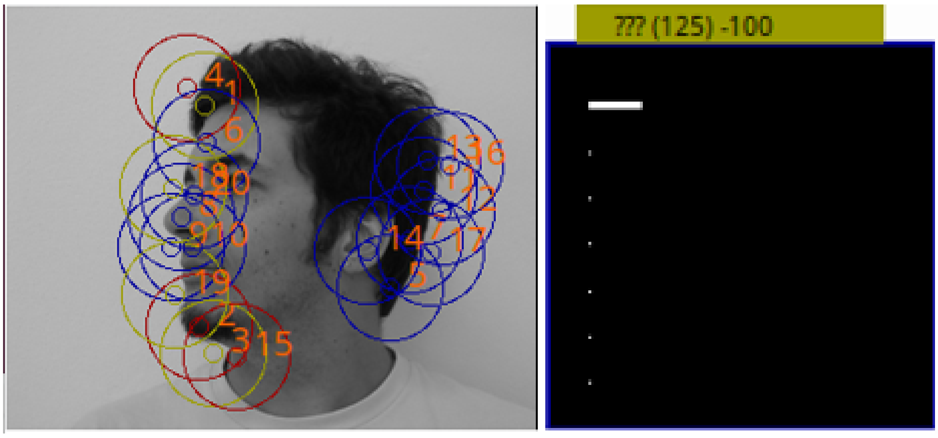
101.png



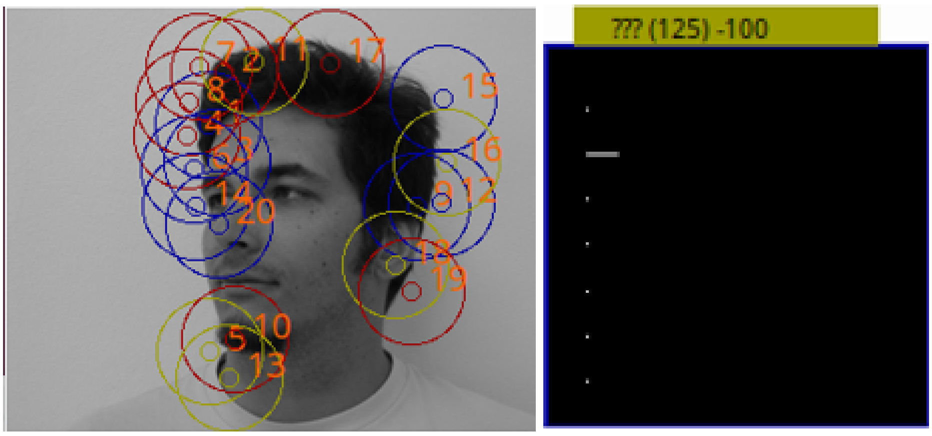
102.png



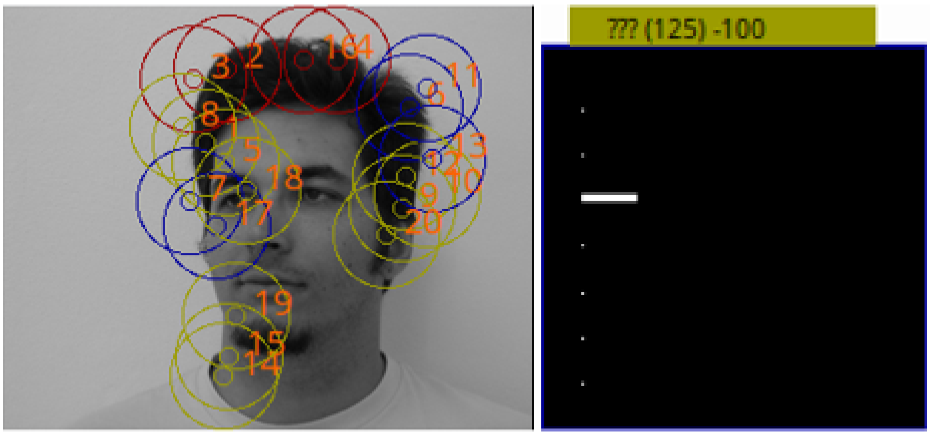
103.png



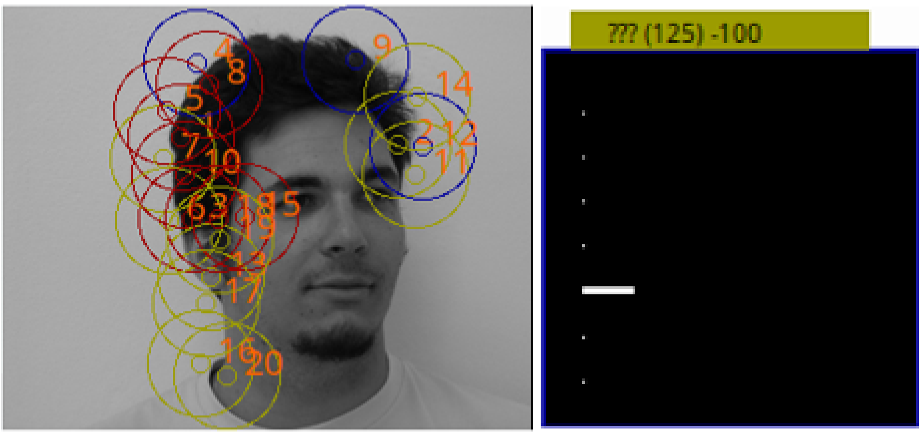
104.png



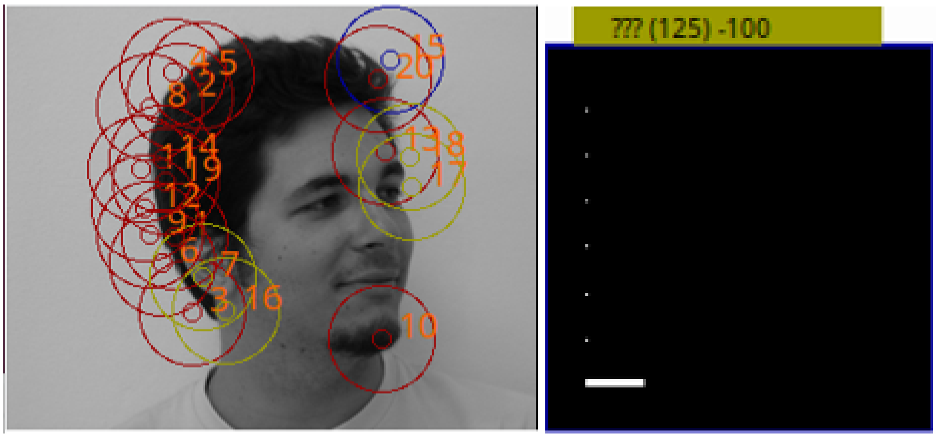
105.png



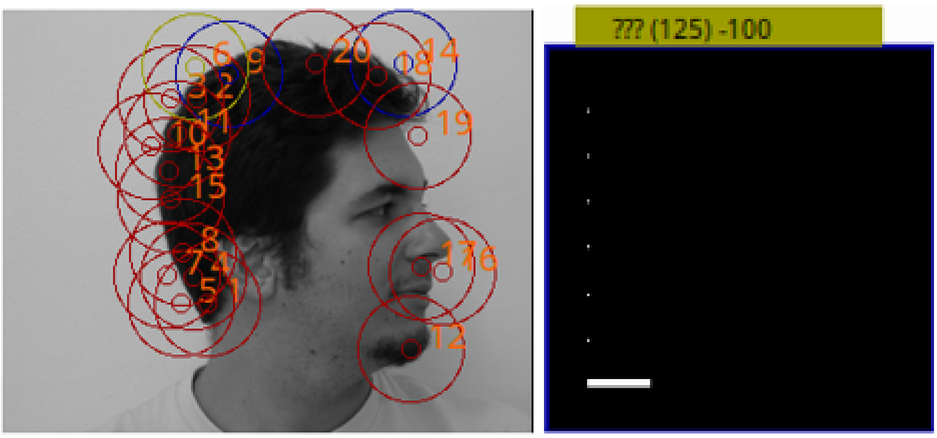
106.png



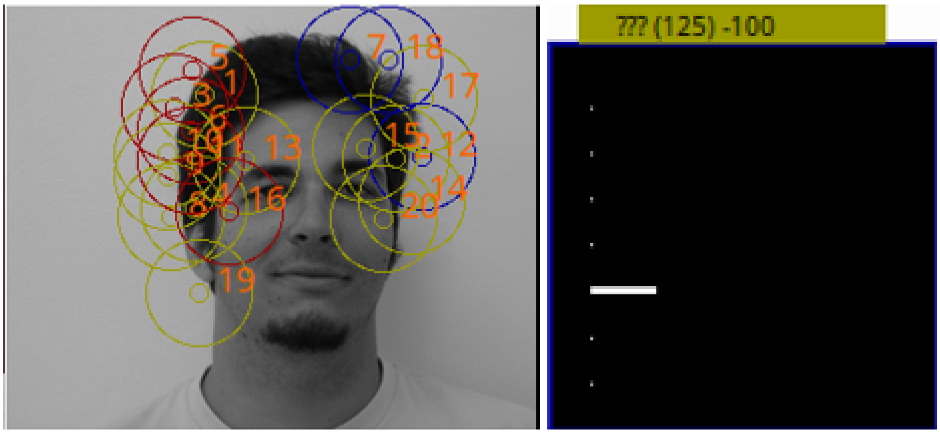
107.png



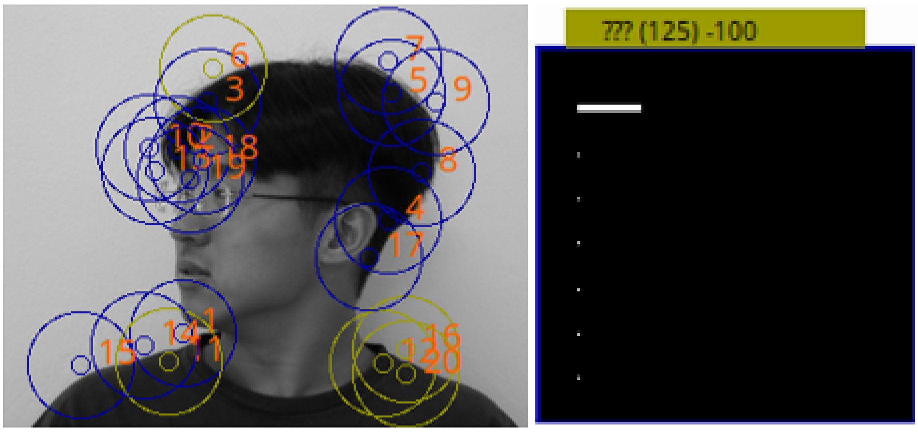
108.png



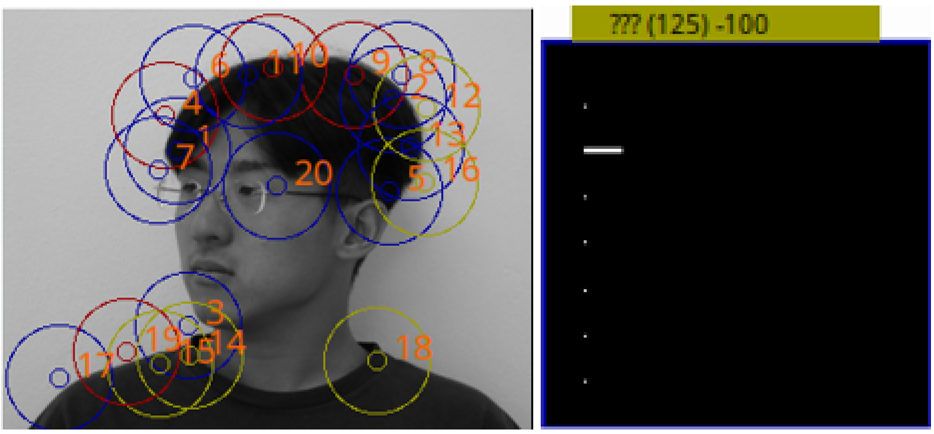
109.png



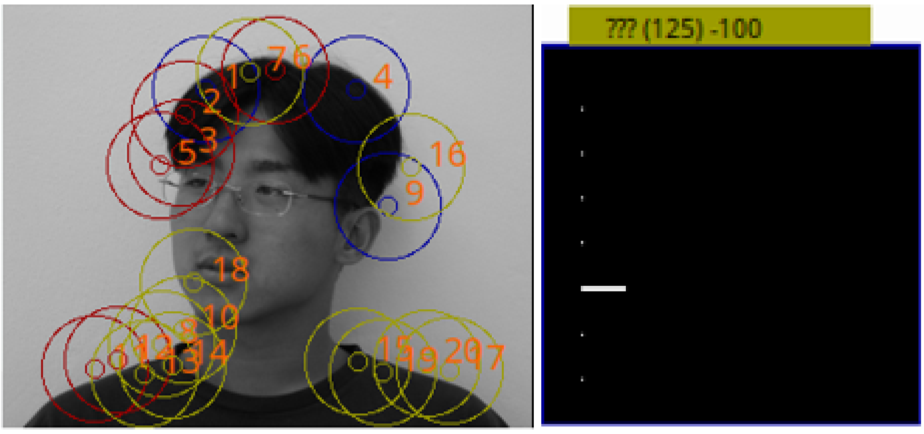
110.png



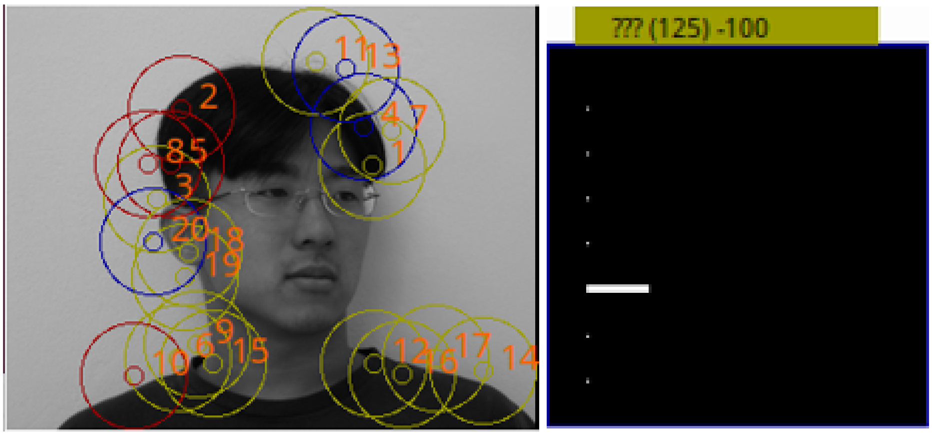
111.png



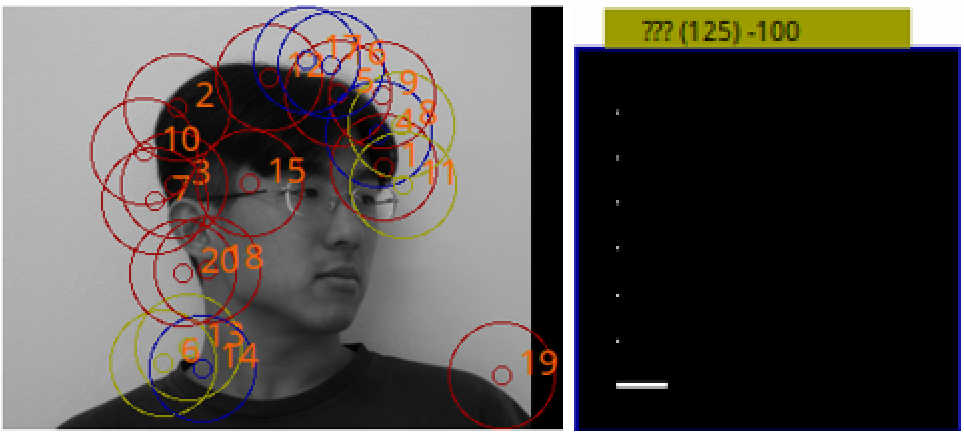
112.png



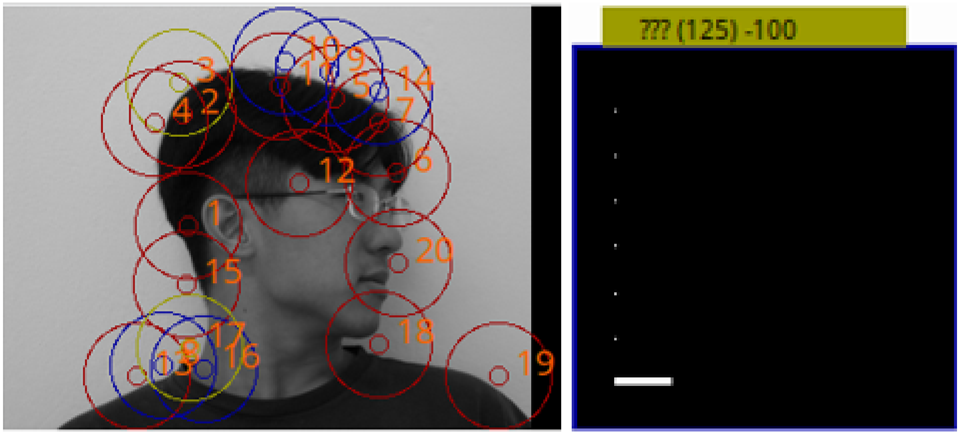
113.png



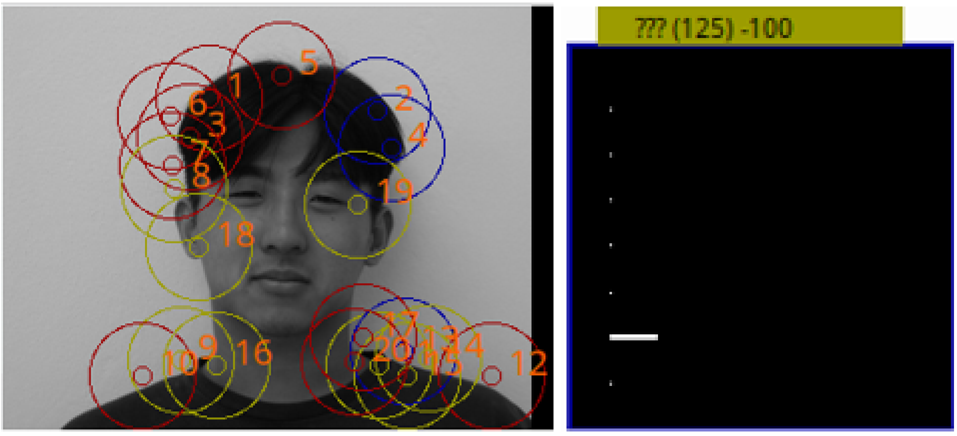
114.png



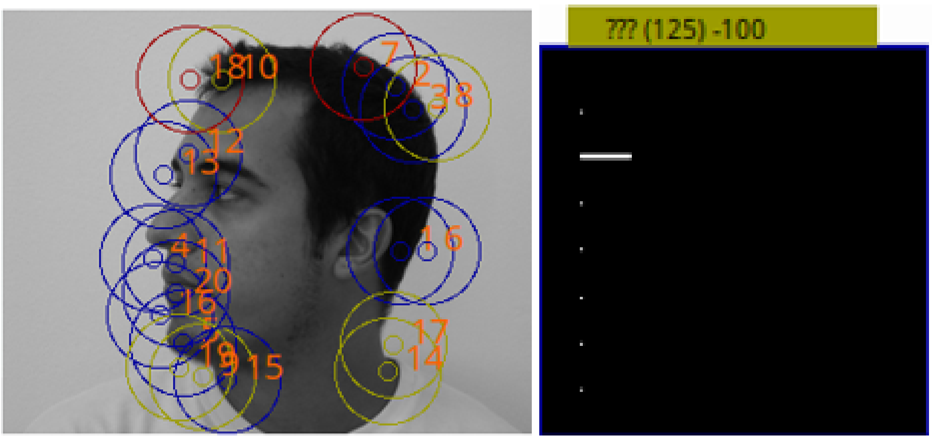
115.png



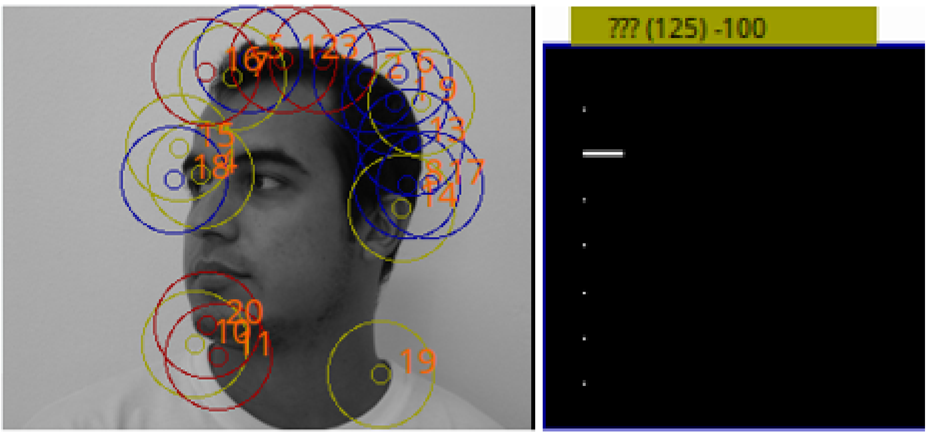
116.png



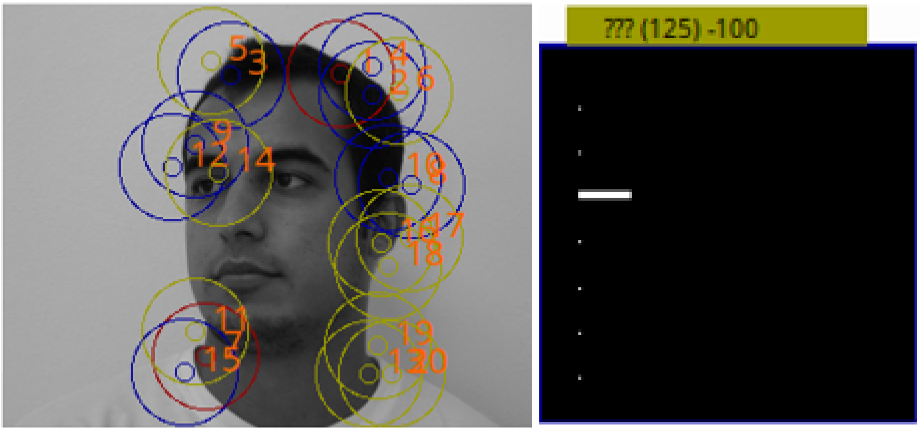
117.png



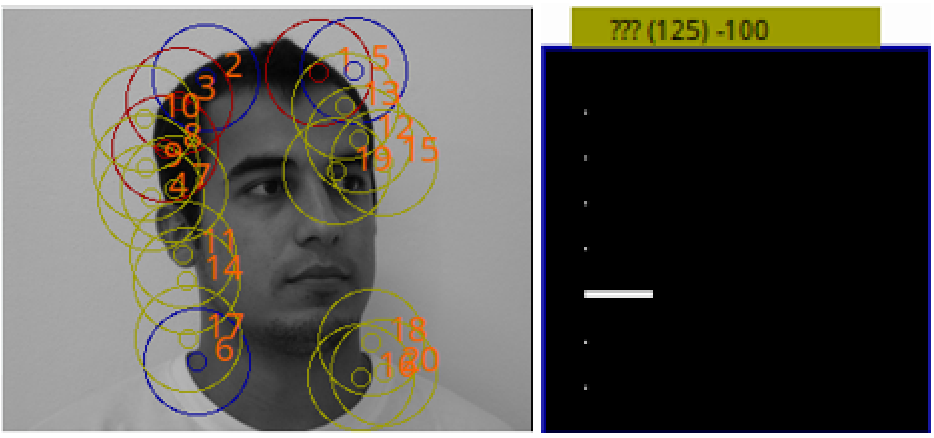
118.png



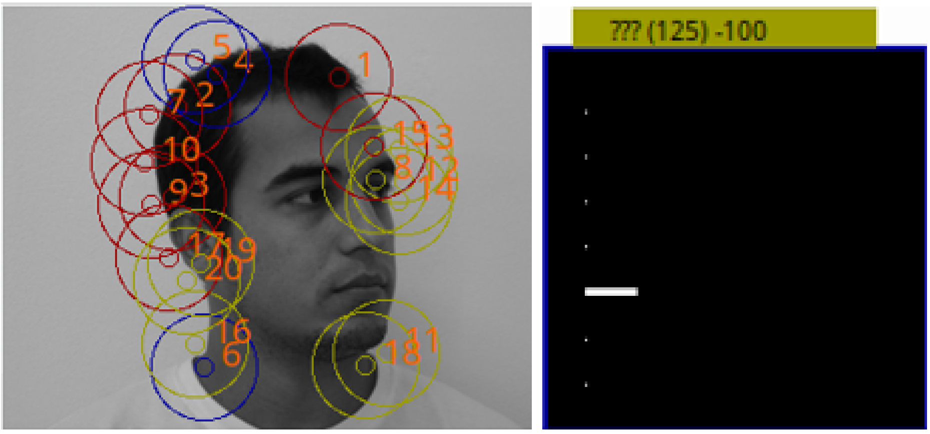
119.png



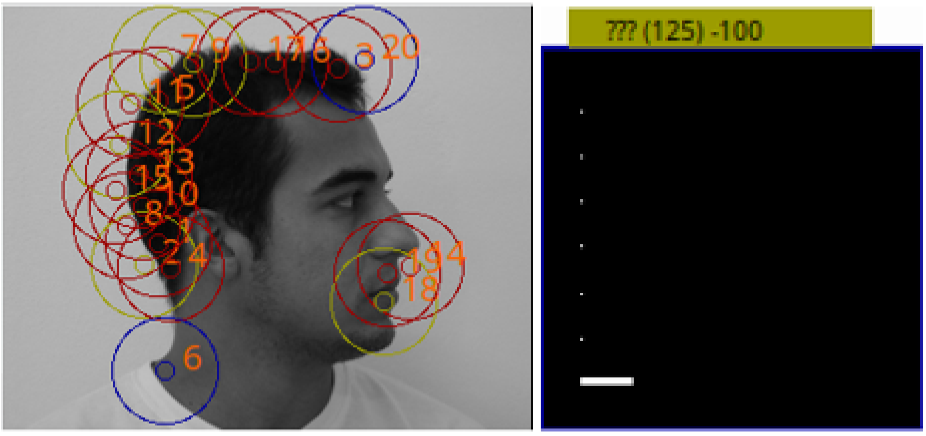
120.png



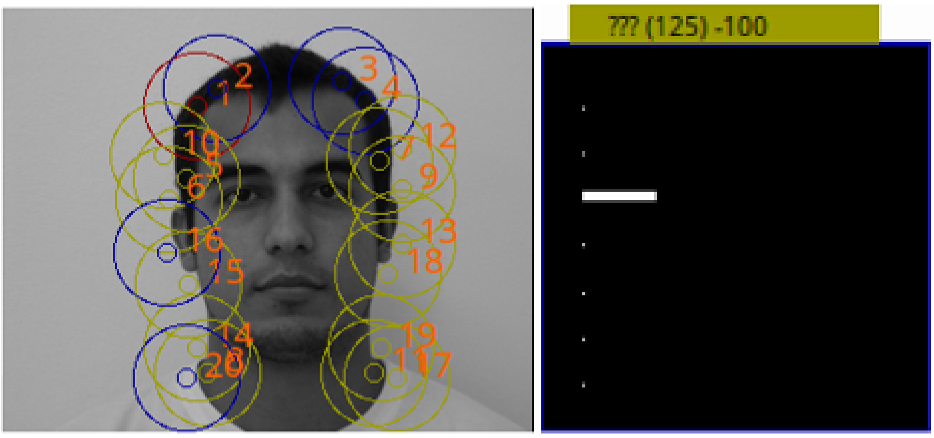
121.png



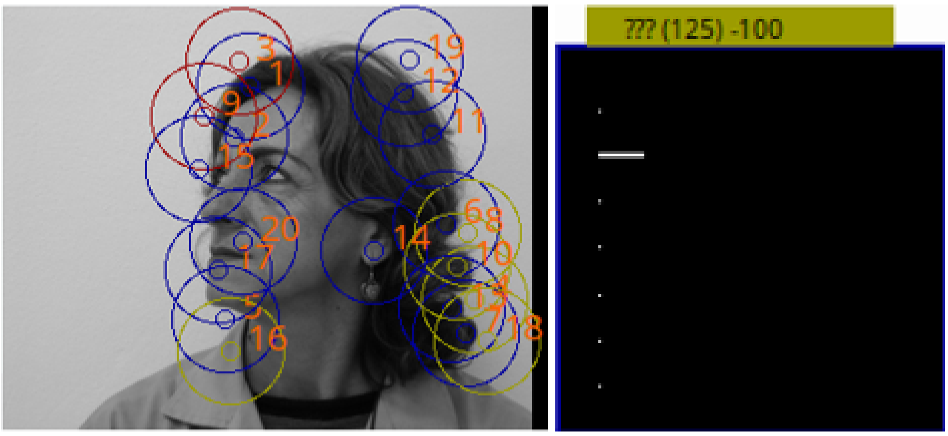
122.png



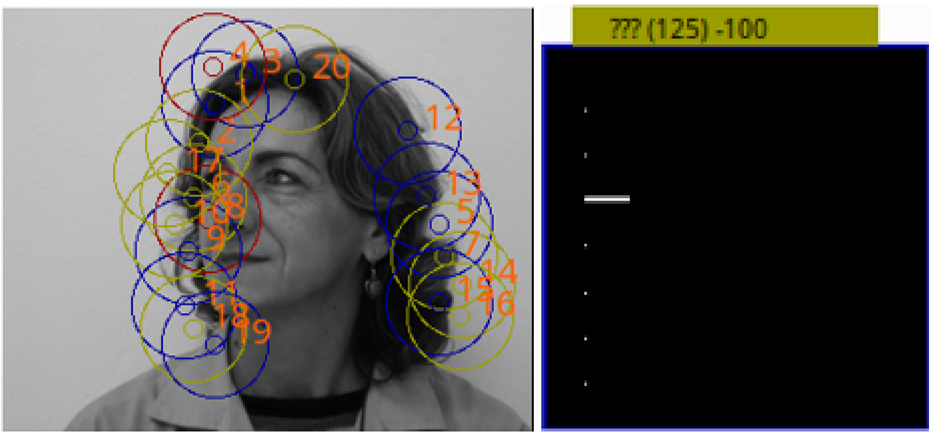
123.png



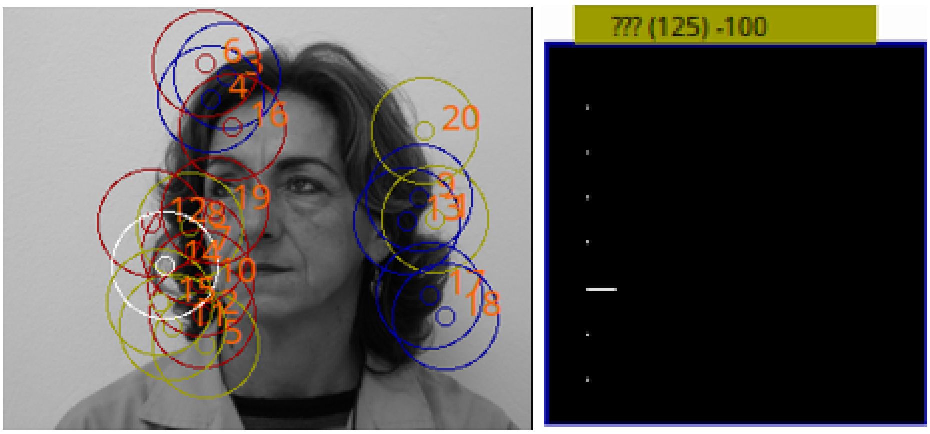
124.png



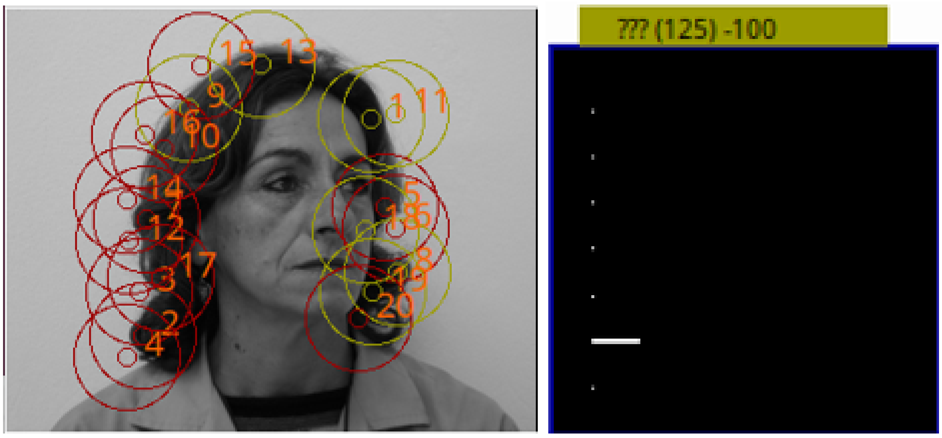
125.png



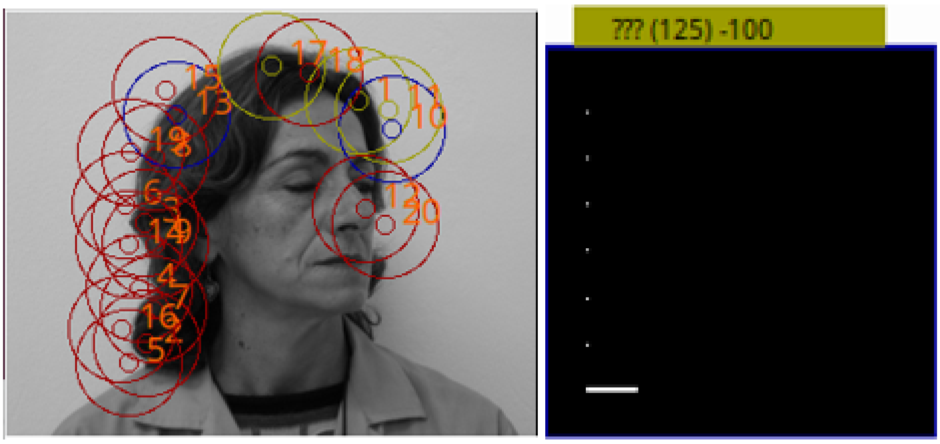
126.png



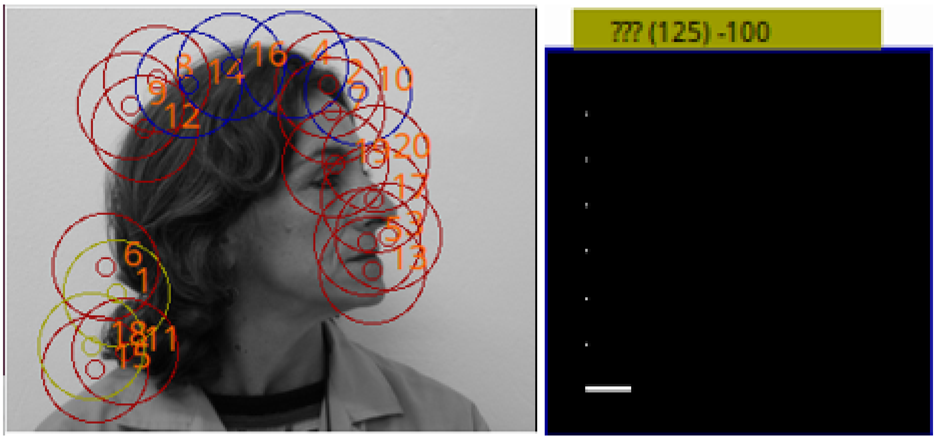
127.png



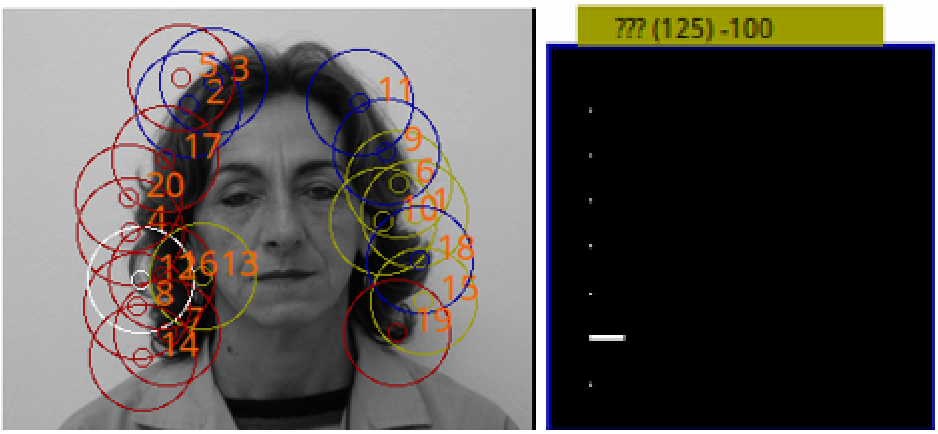
128.png



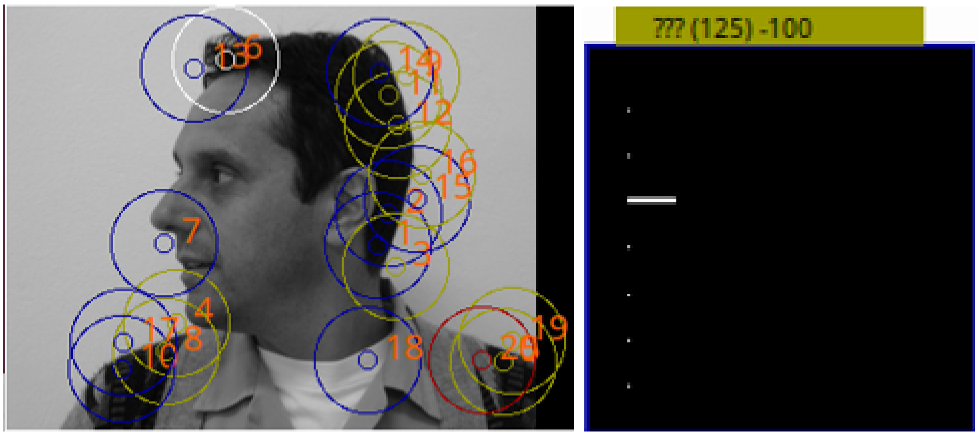
129.png



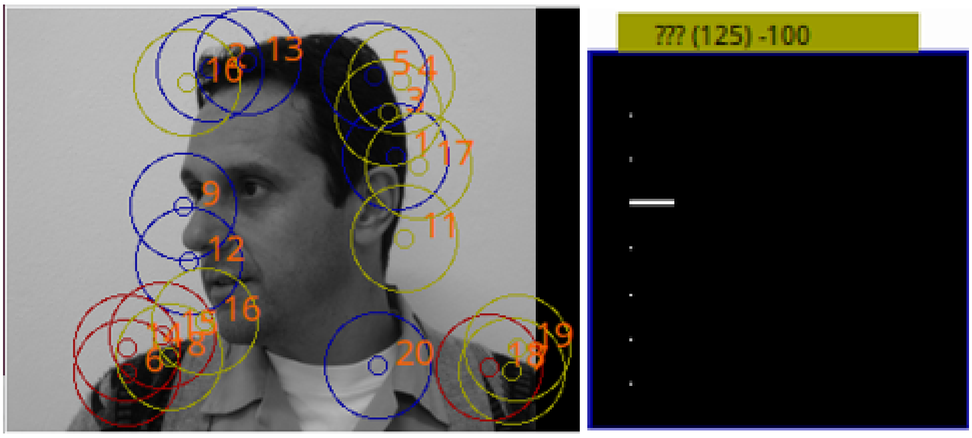
130.png



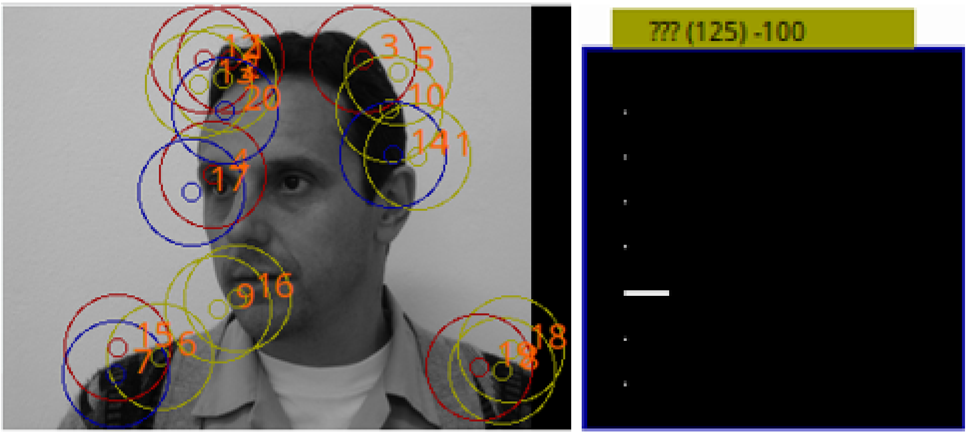
131.png



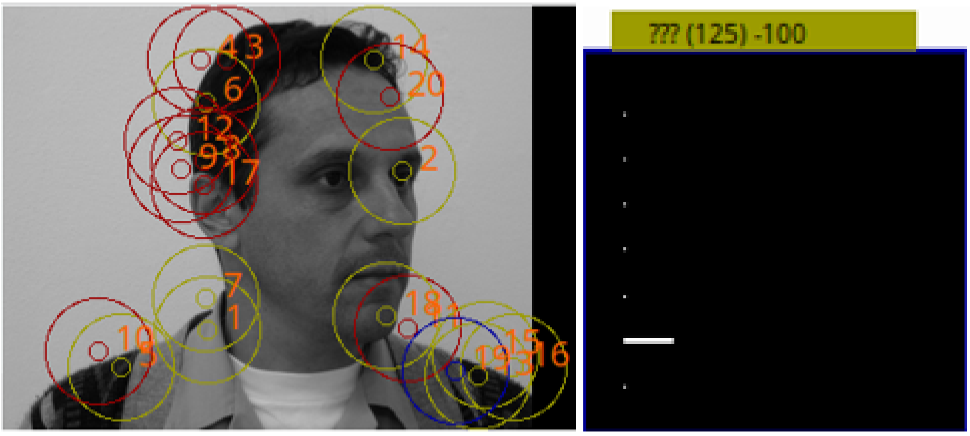
132.png



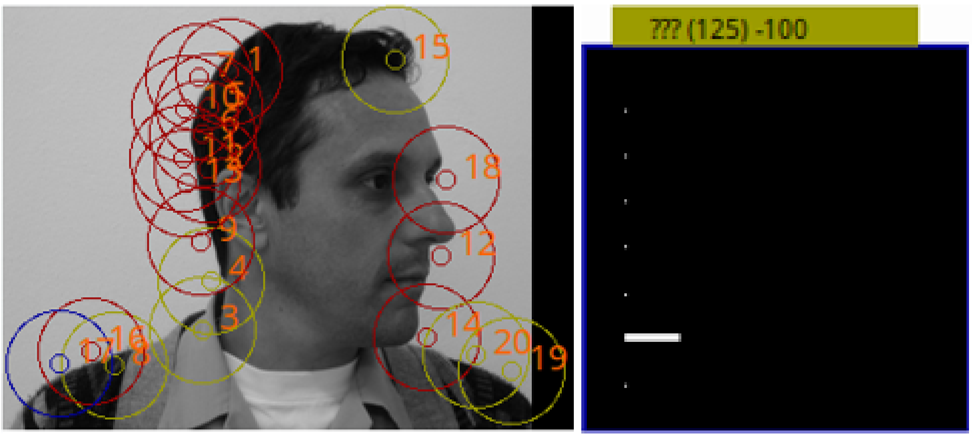
133.png



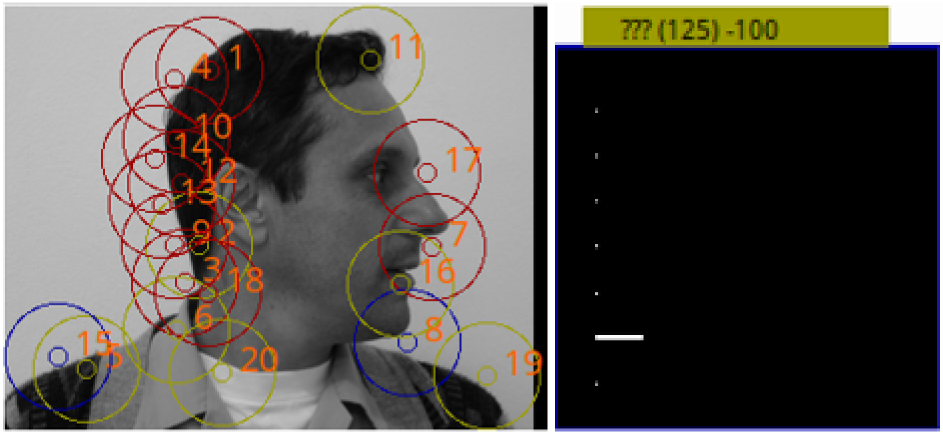
134.png



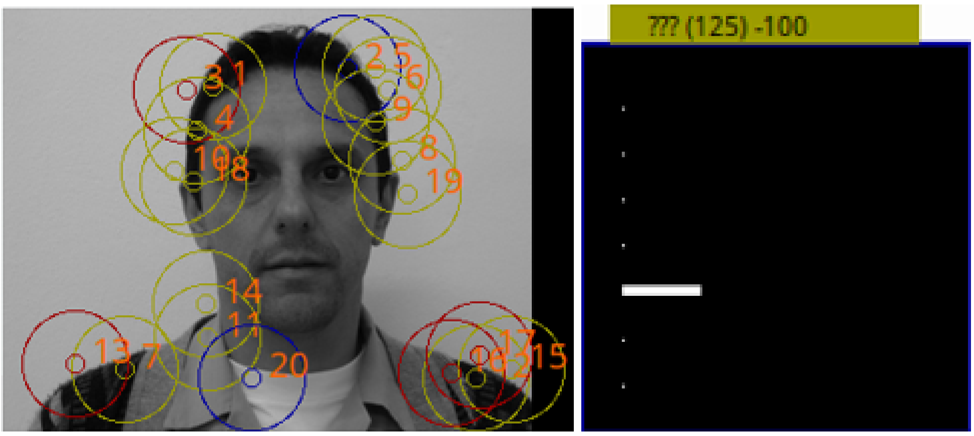
135.png



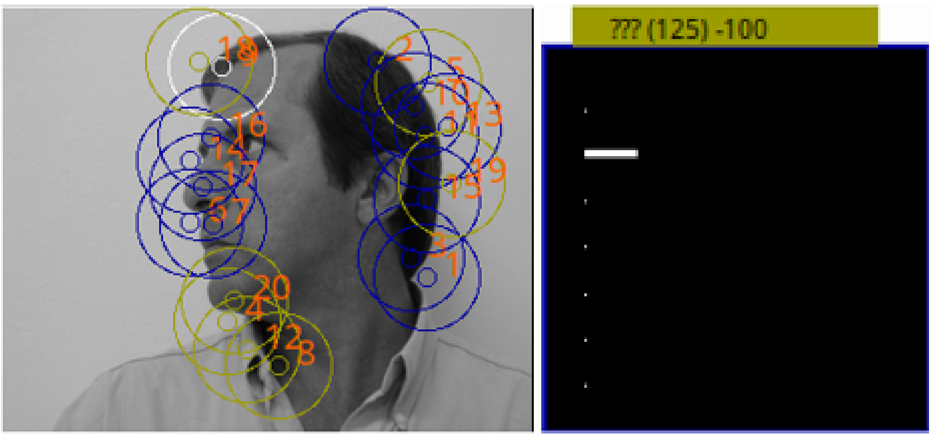
136.png



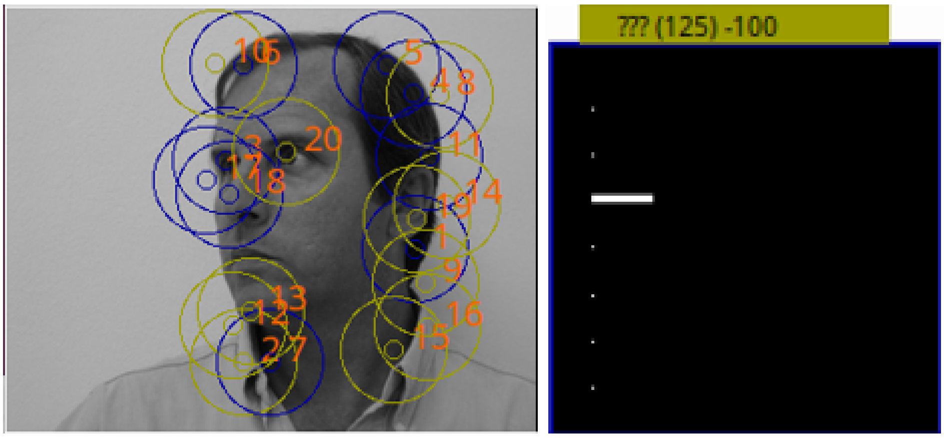
137.png



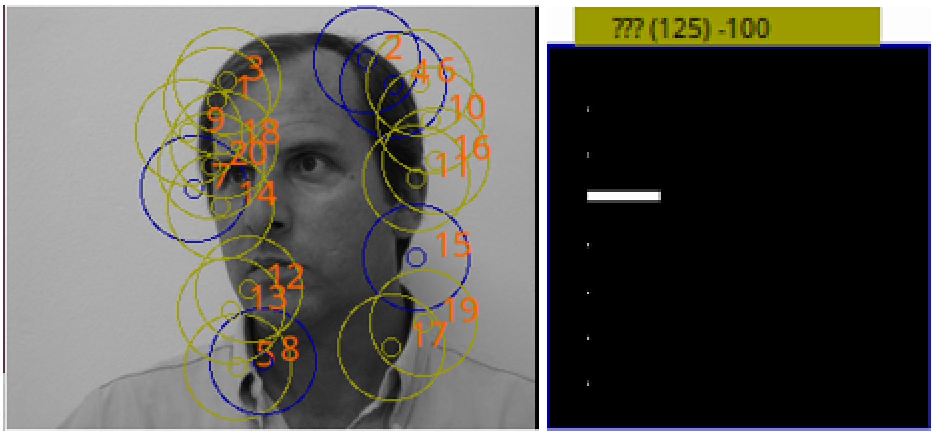
138.png



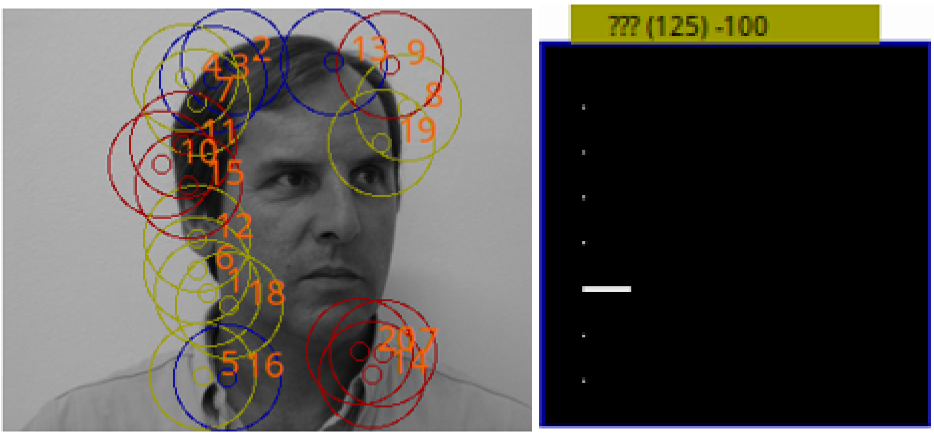
139.png



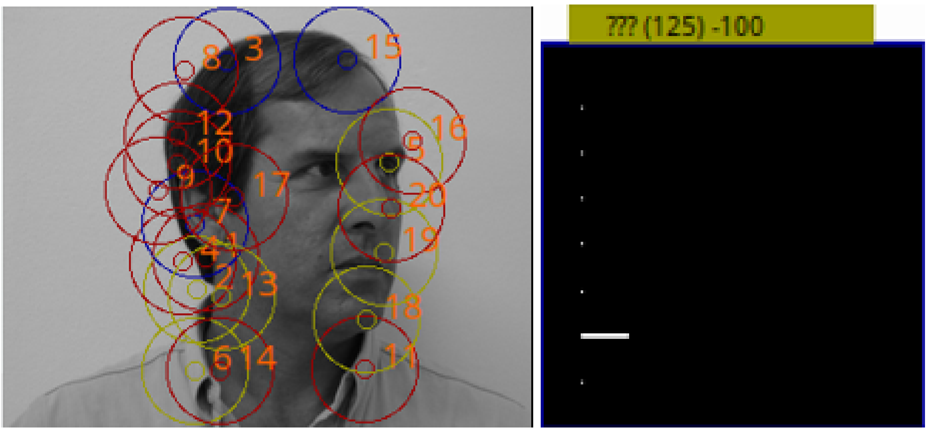
140.png



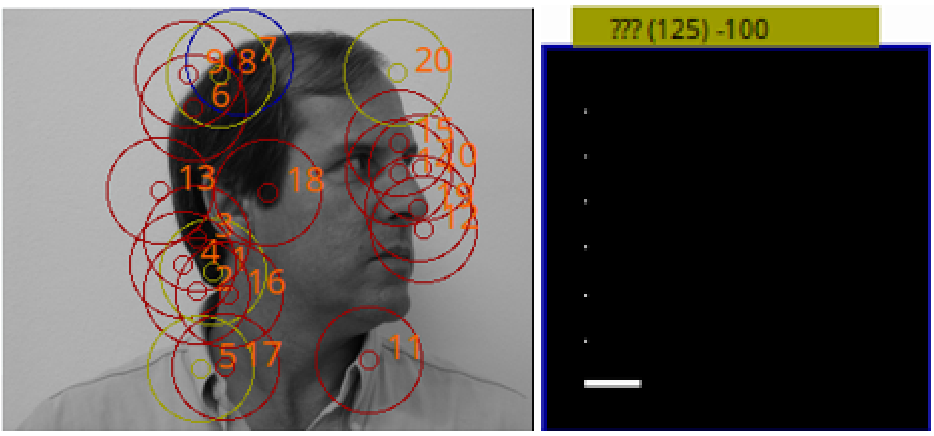
141.png



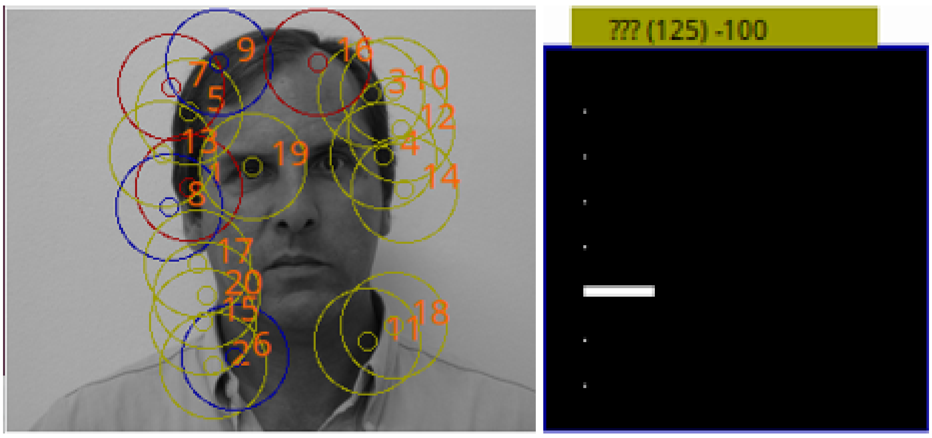
142.png



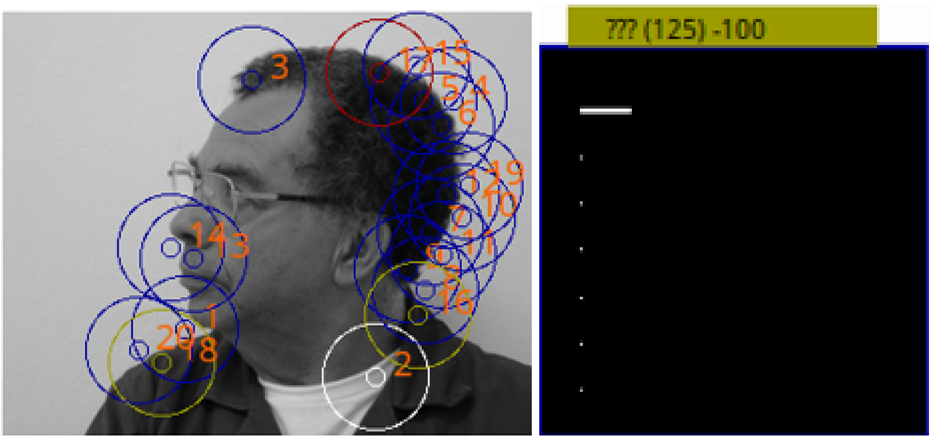
143.png



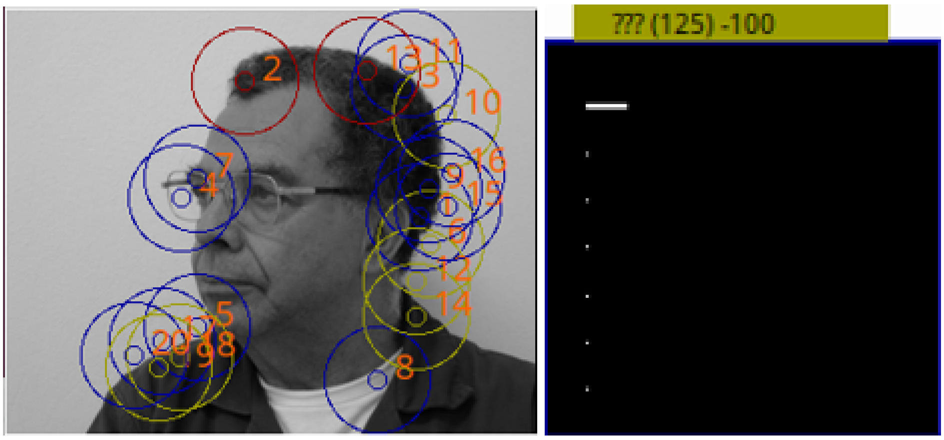
144.png



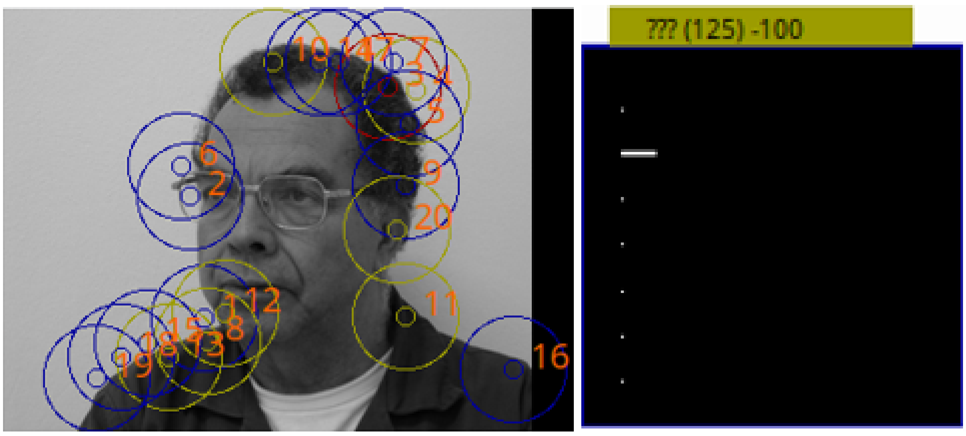
145.png



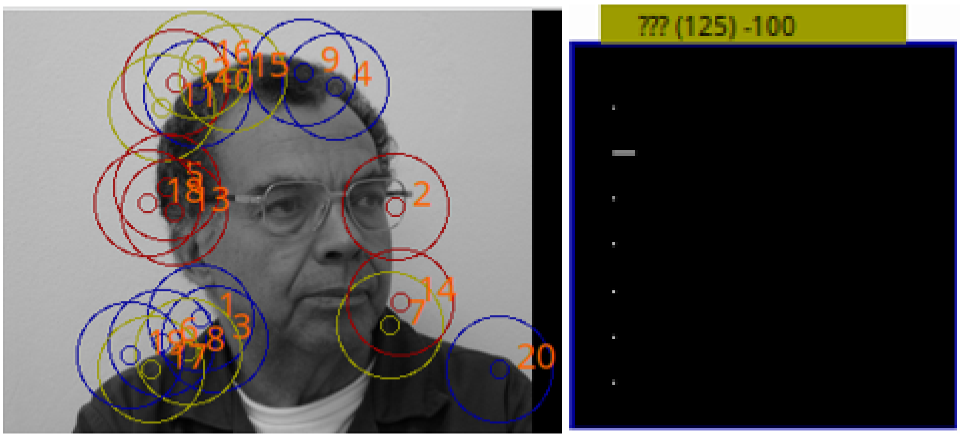
146.png



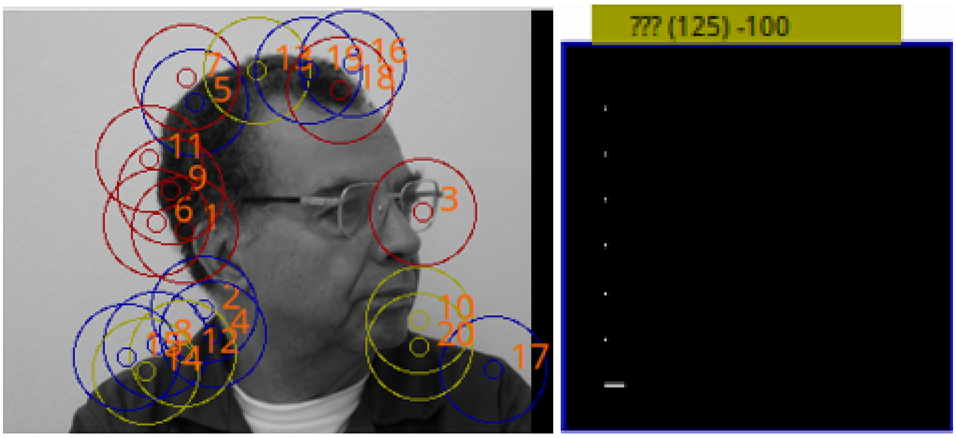
147.png



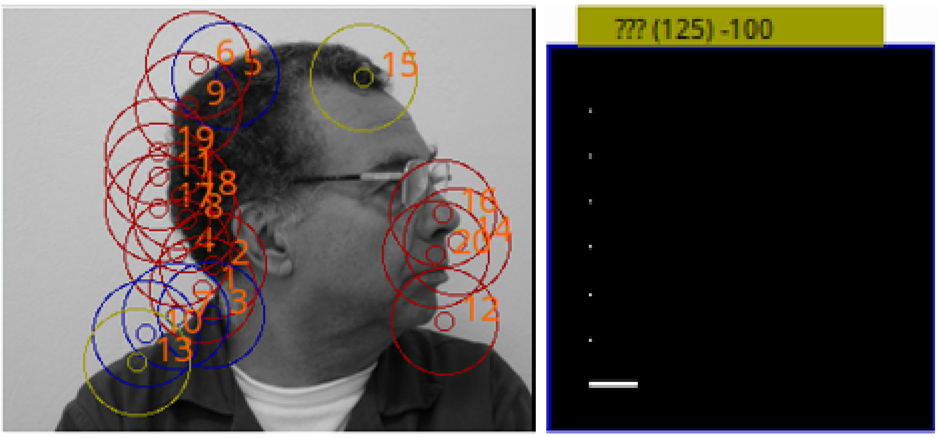
148.png



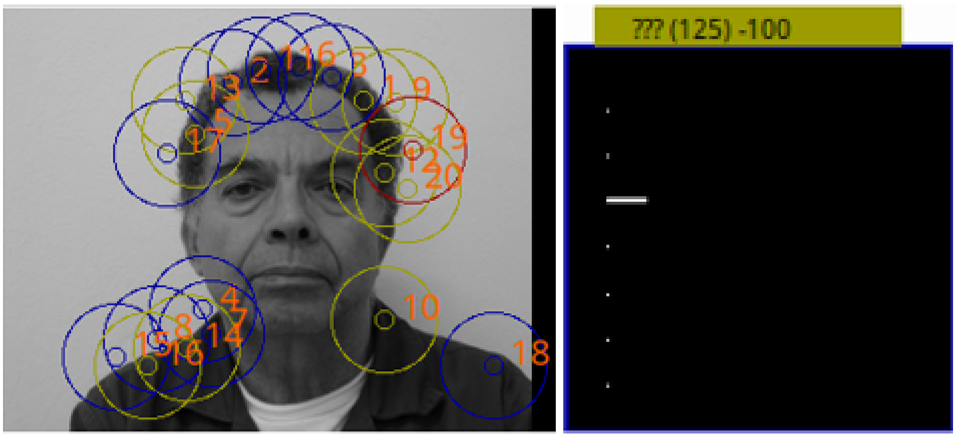
149.png



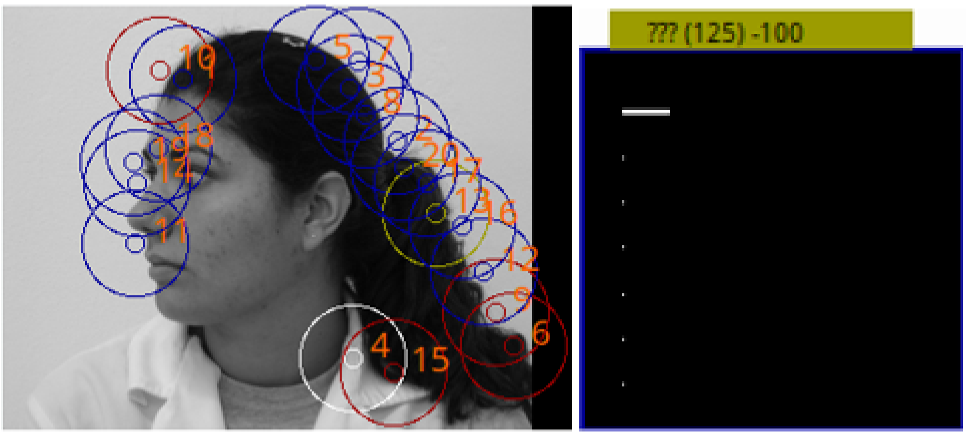
150.png



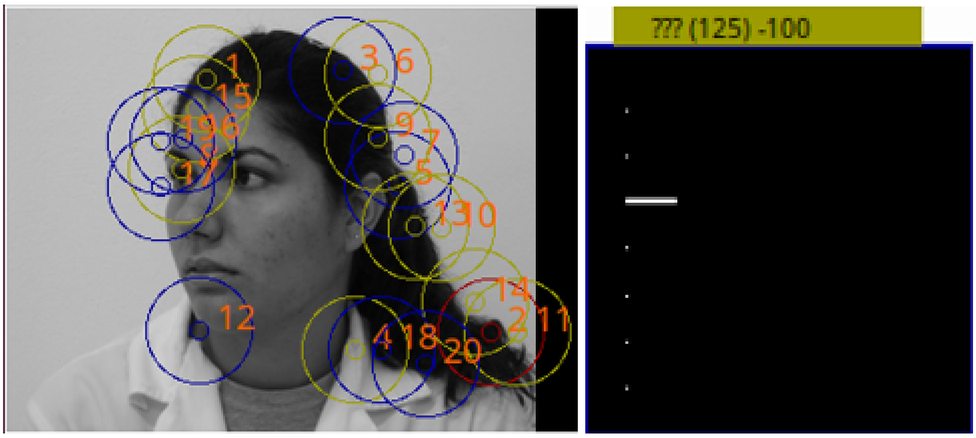
151.png



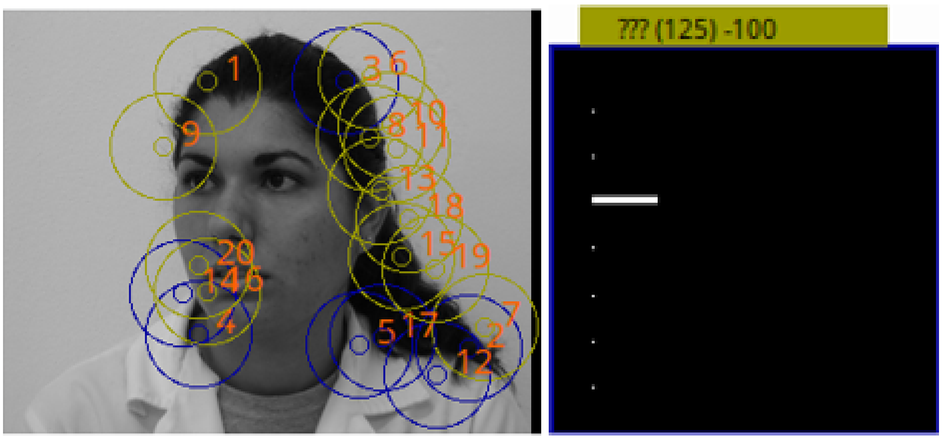
152.png



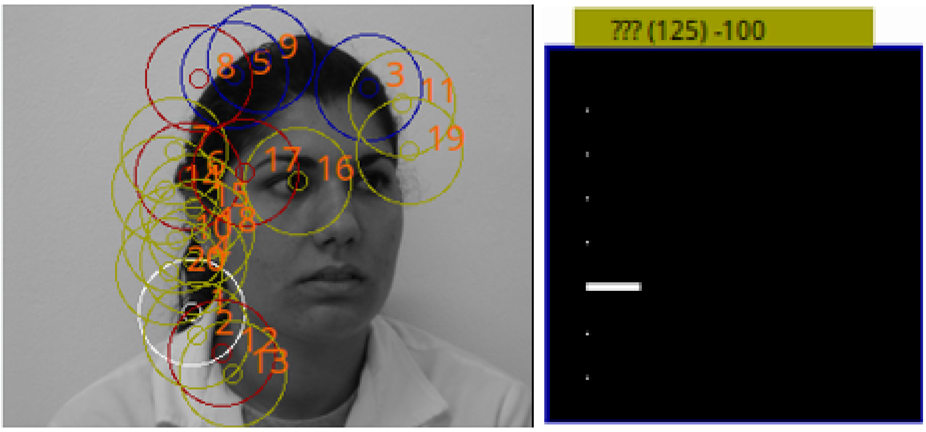
153.png



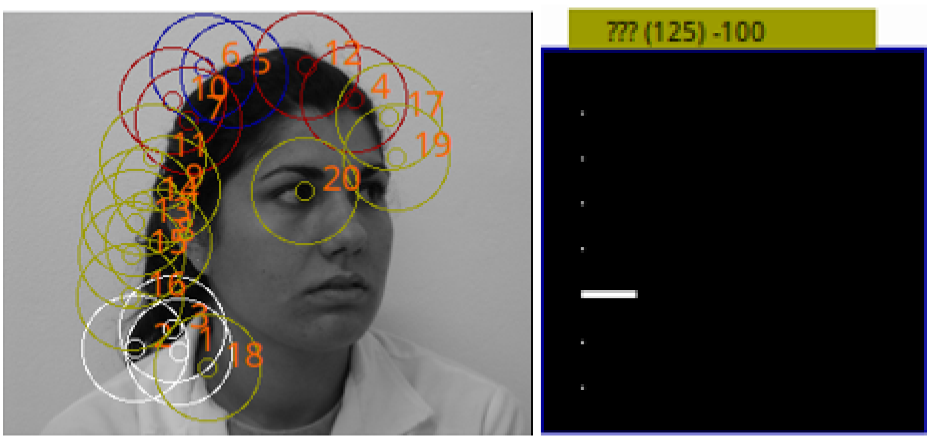
154.png



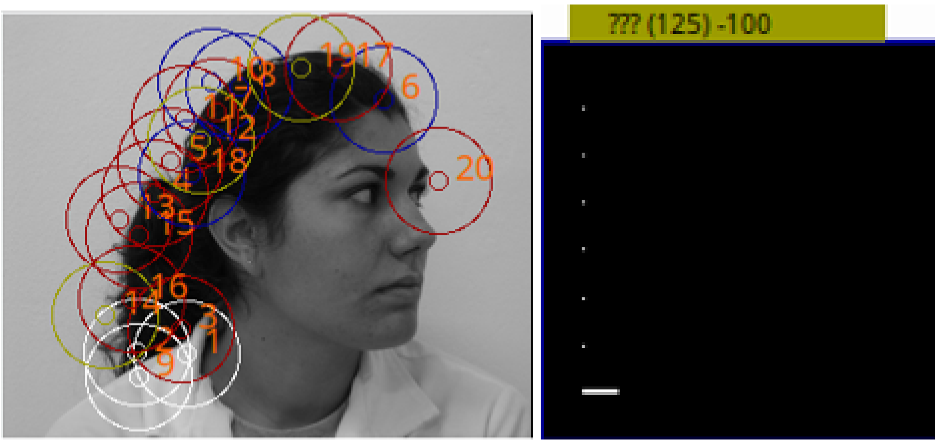
155.png



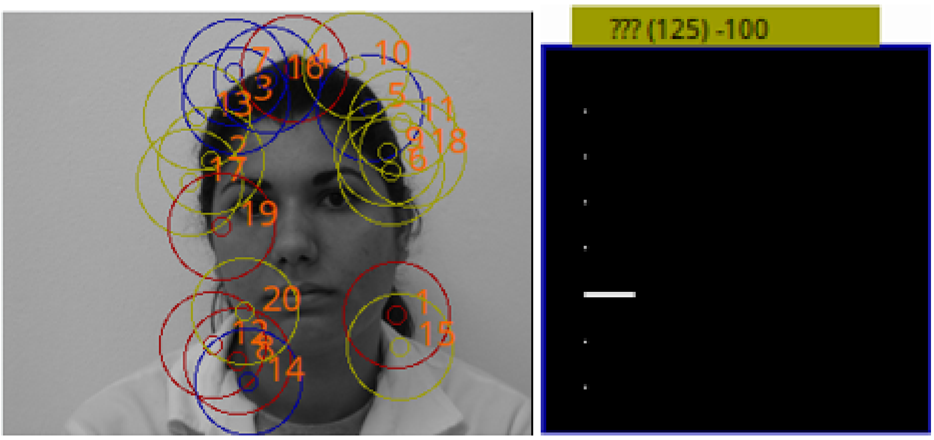
156.png



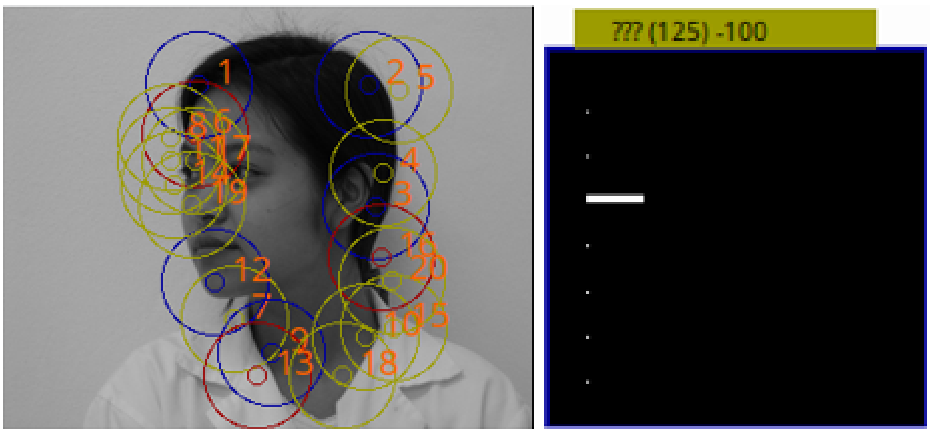
157.png



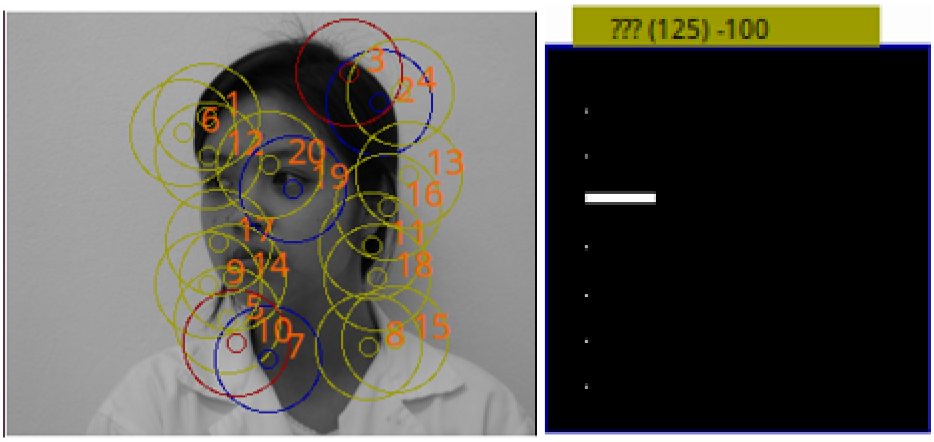
158.png



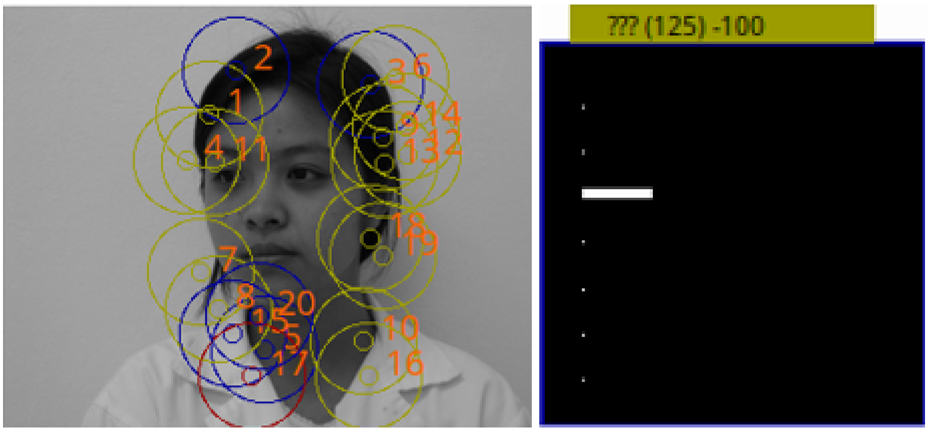
159.png



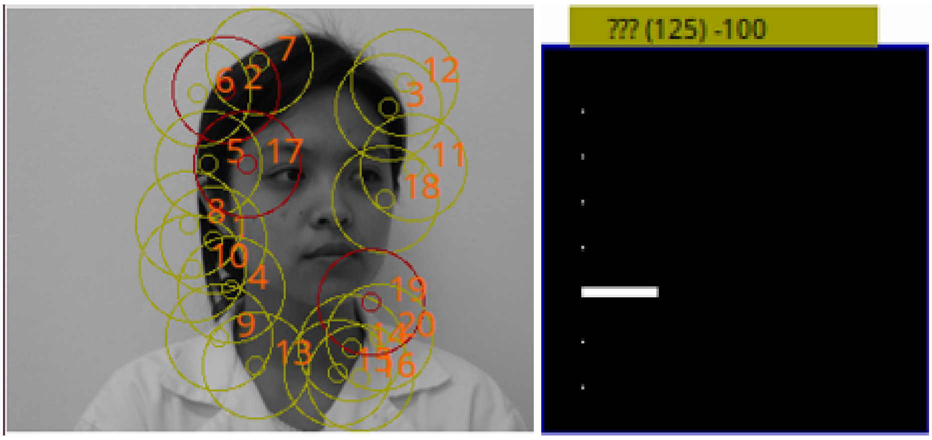
160.png



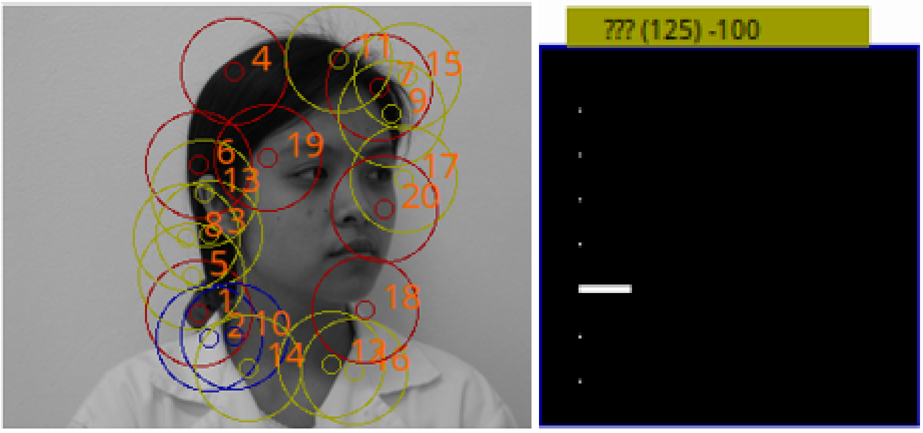
161.png



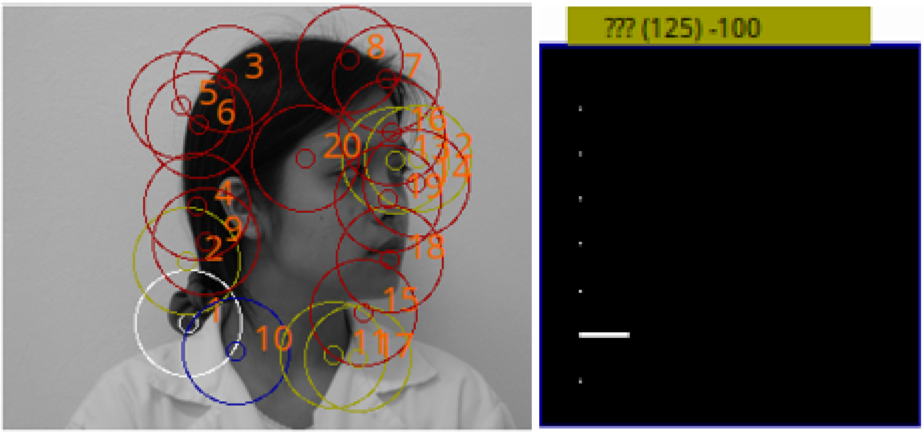
162.png



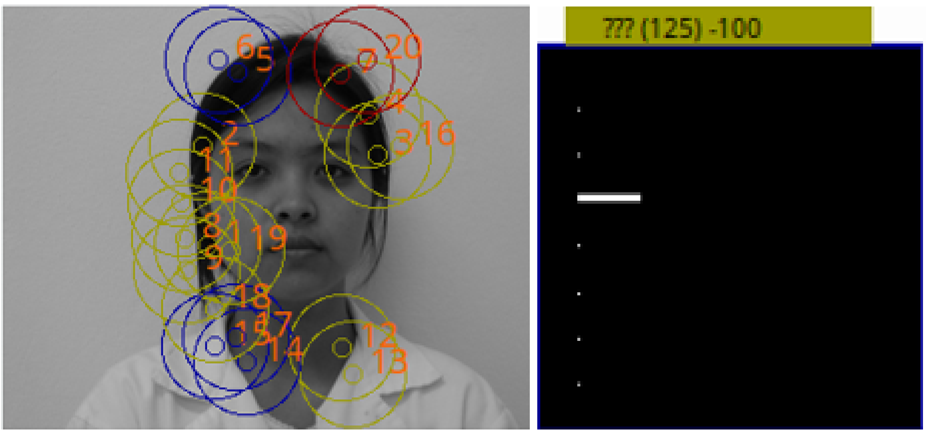
163.png



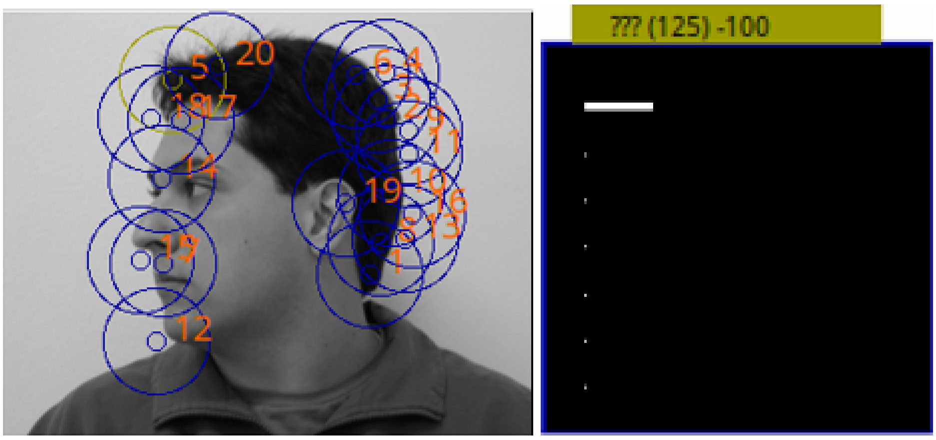
164.png



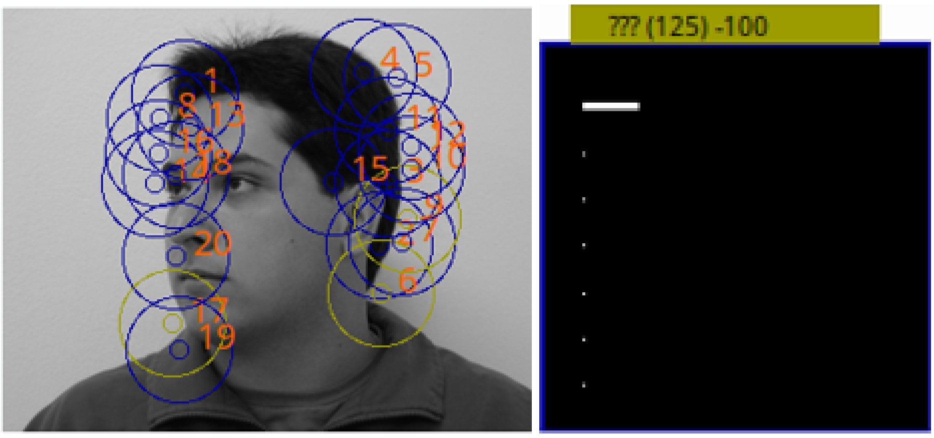
165.png



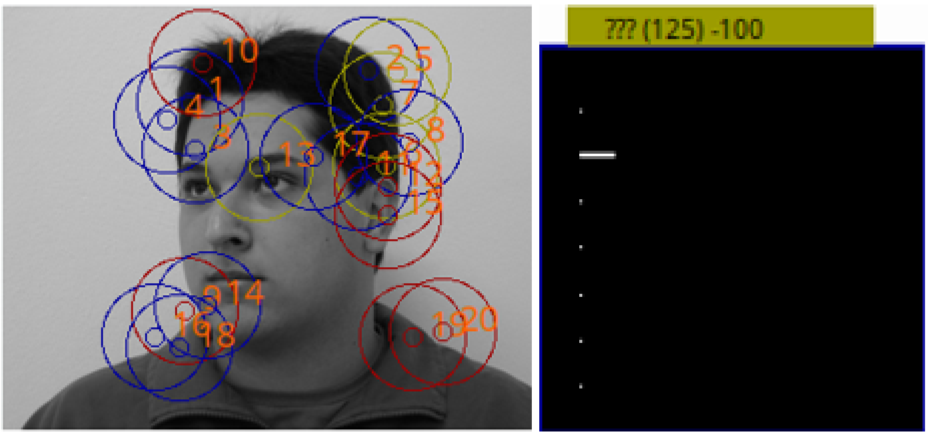
166.png



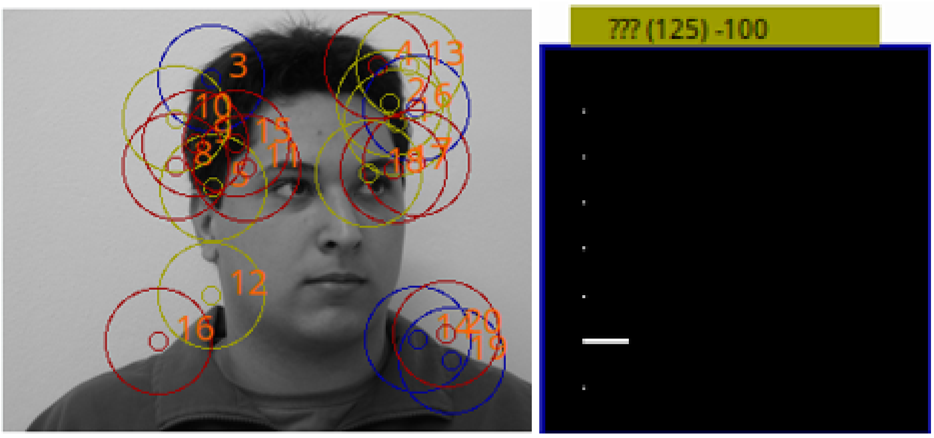
167.png



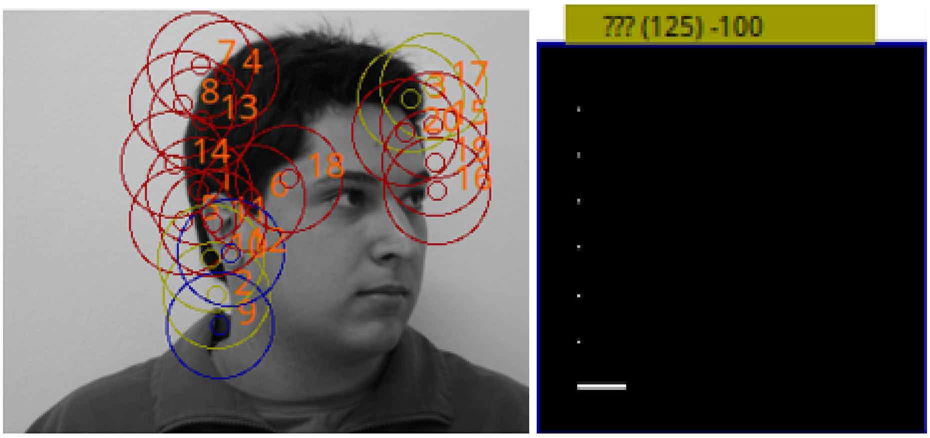
168.png



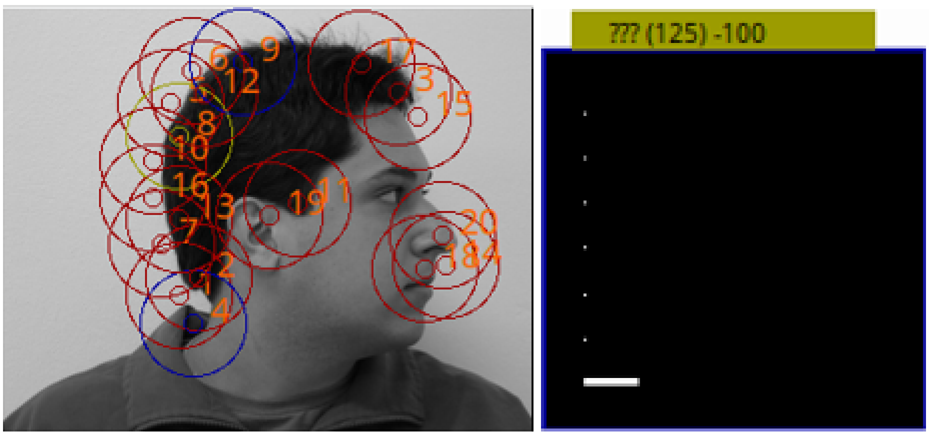
169.png



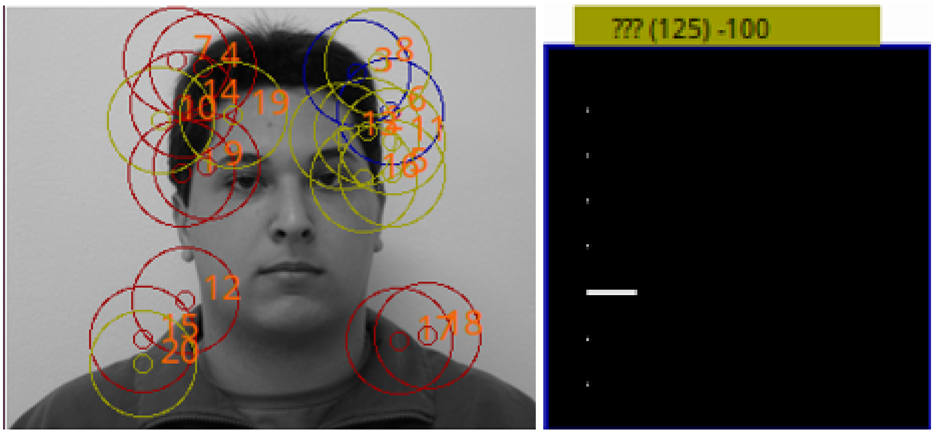
170.png



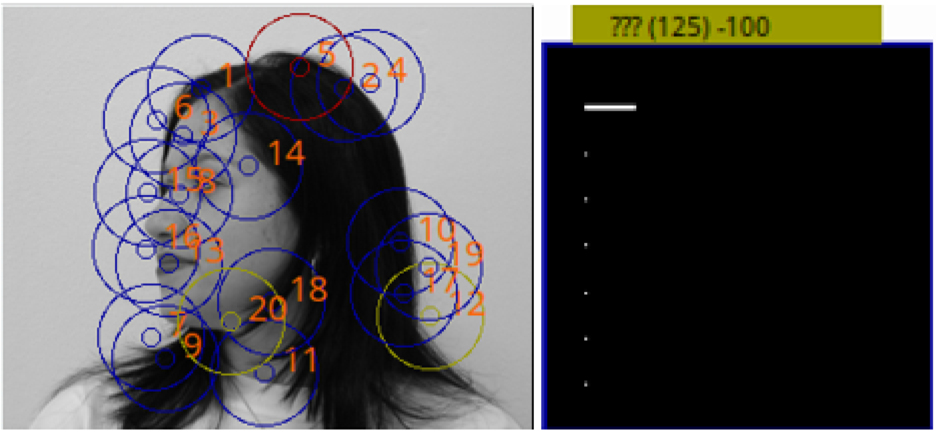
171.png



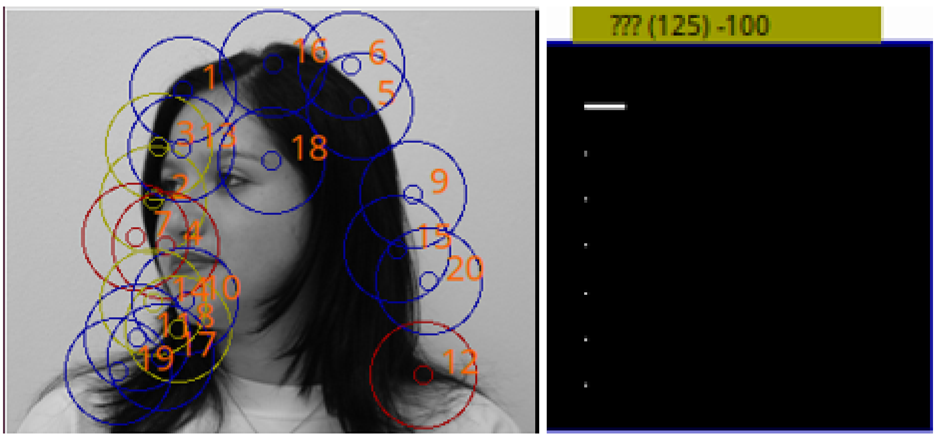
172.png



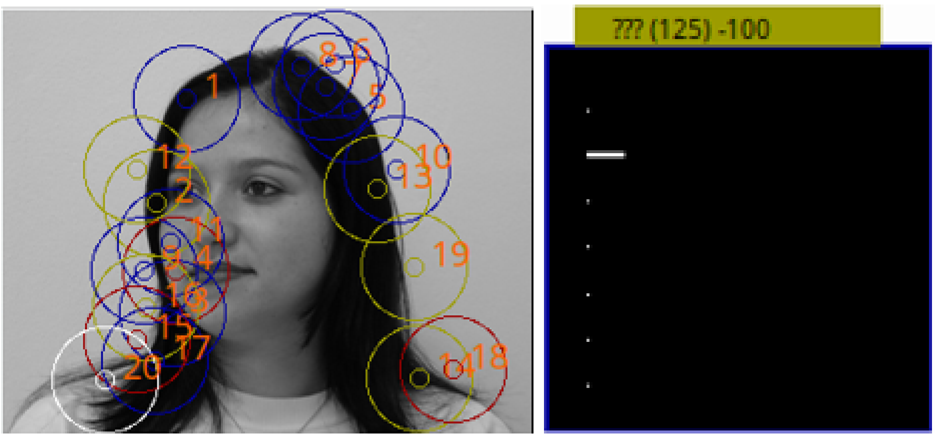
173.png



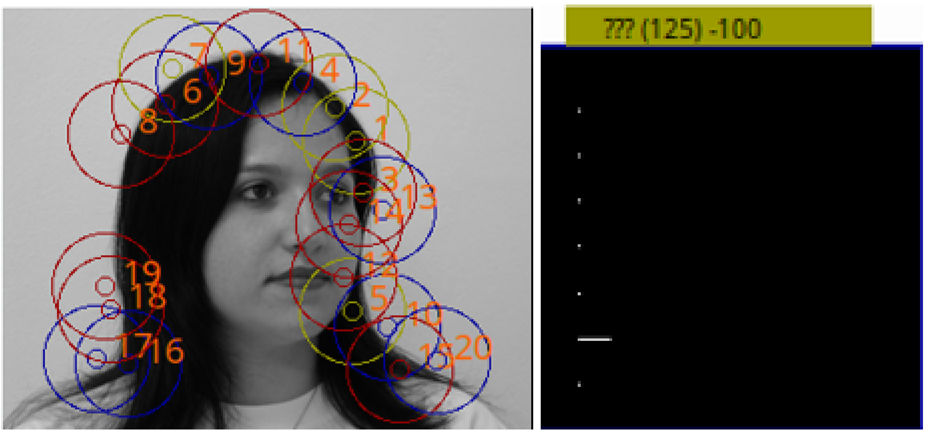
174.png



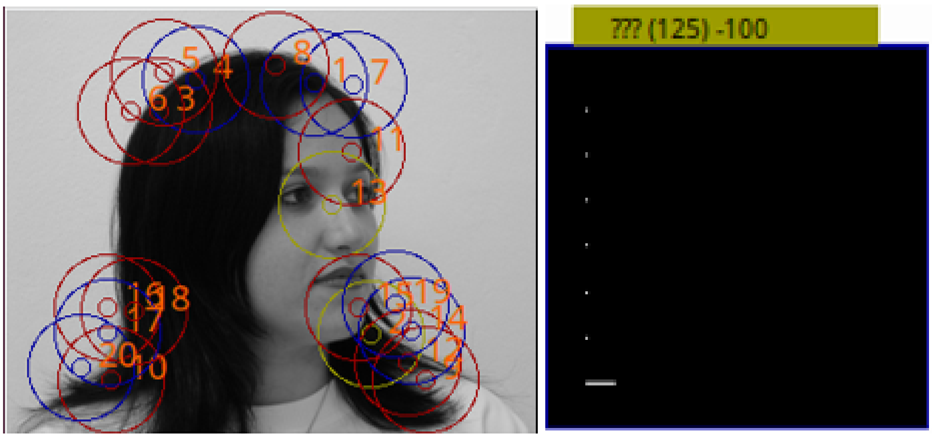
175.png



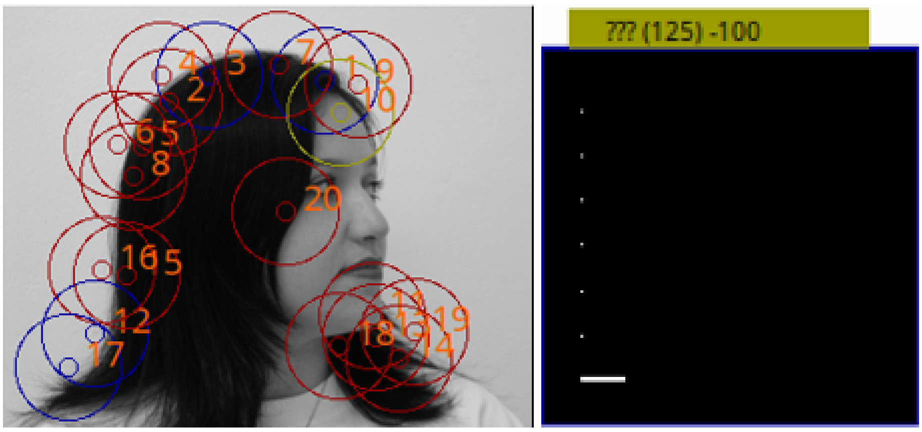
176.png



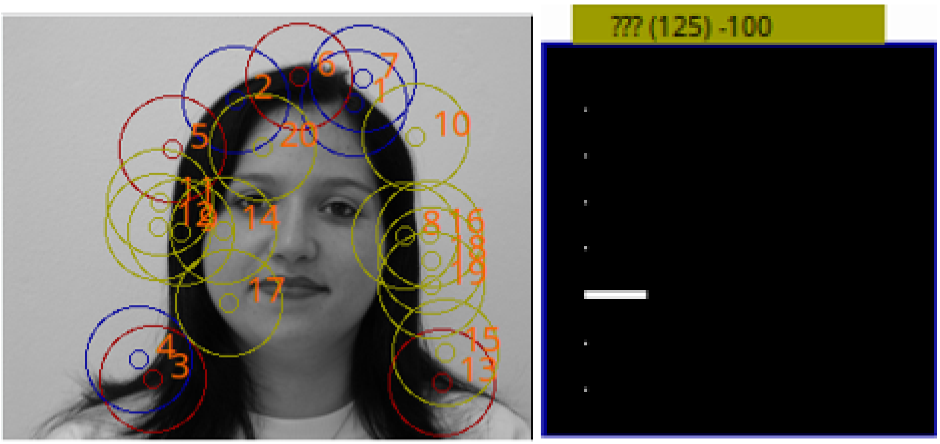
177.png



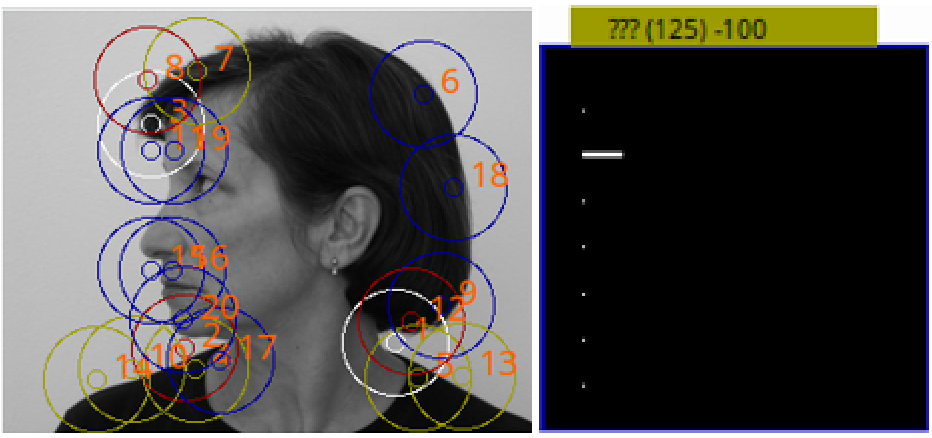
178.png



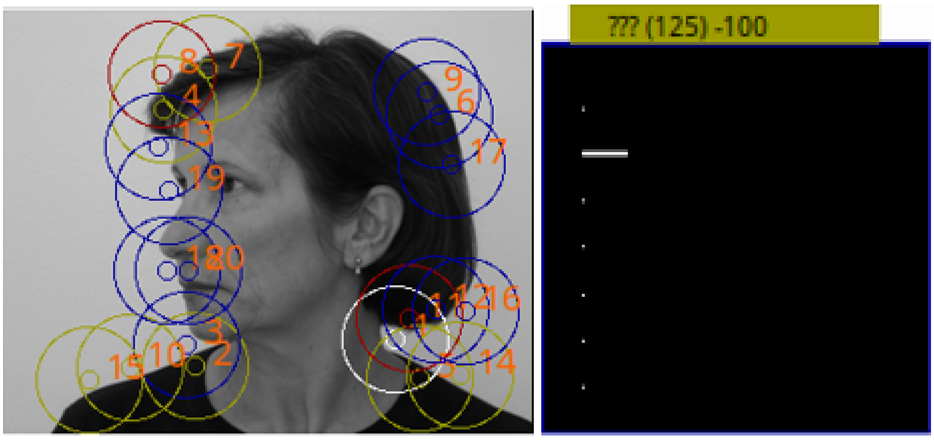
179.png



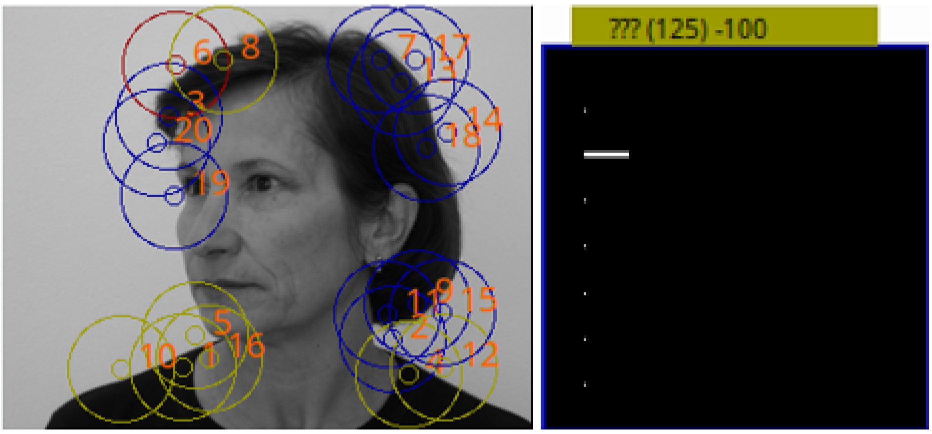
180.png



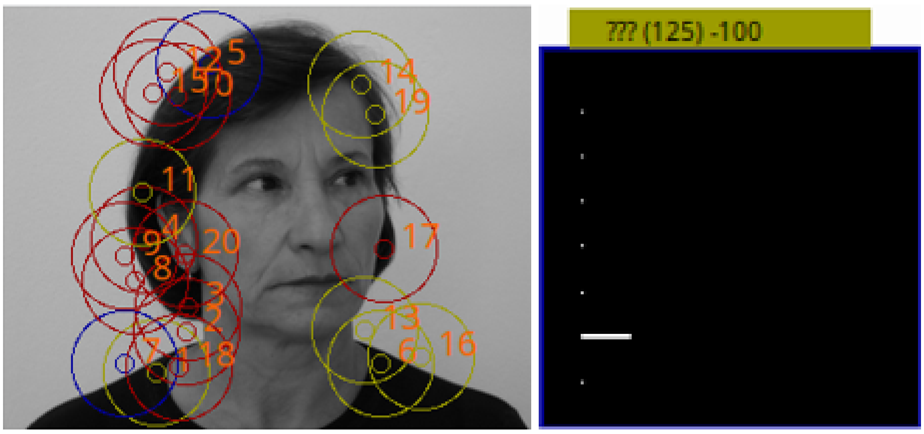
181.png



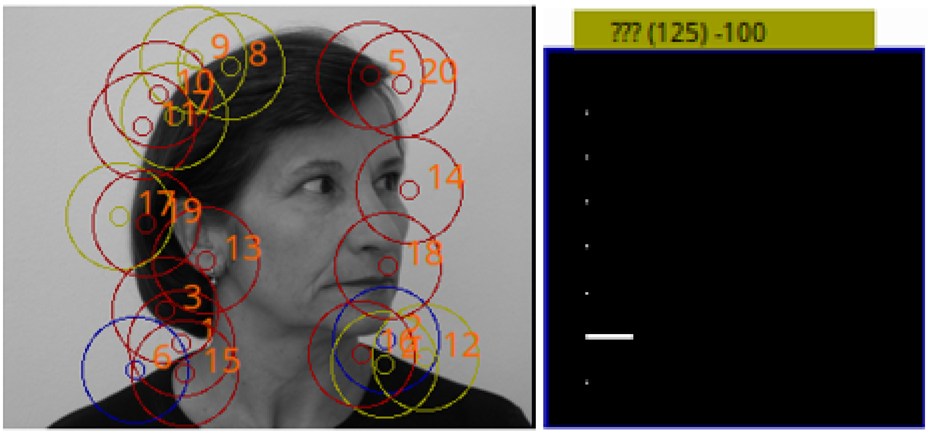
182.png



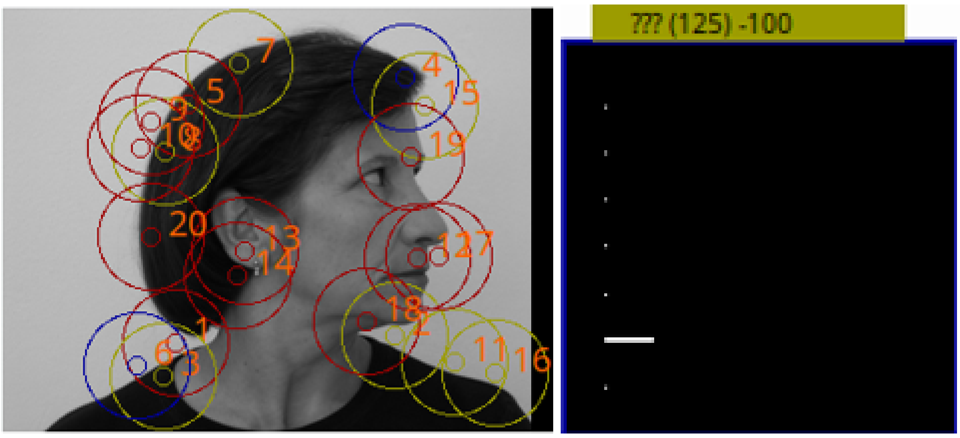
183.png



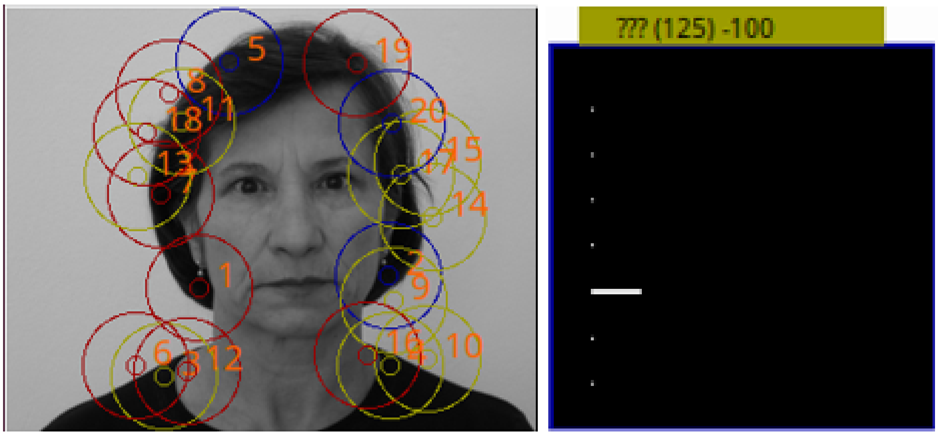
184.png



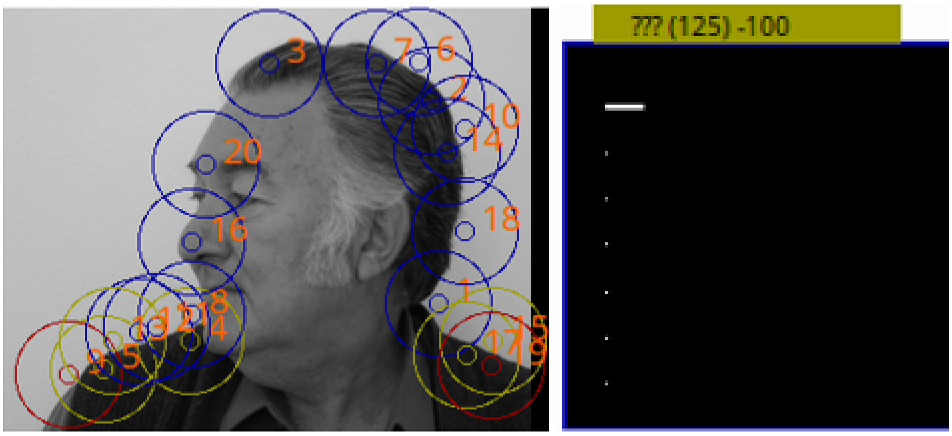
185.png



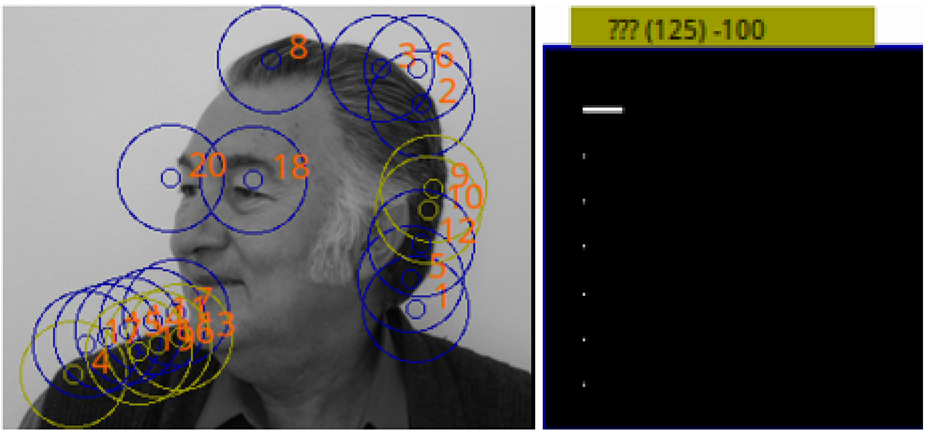
186.png



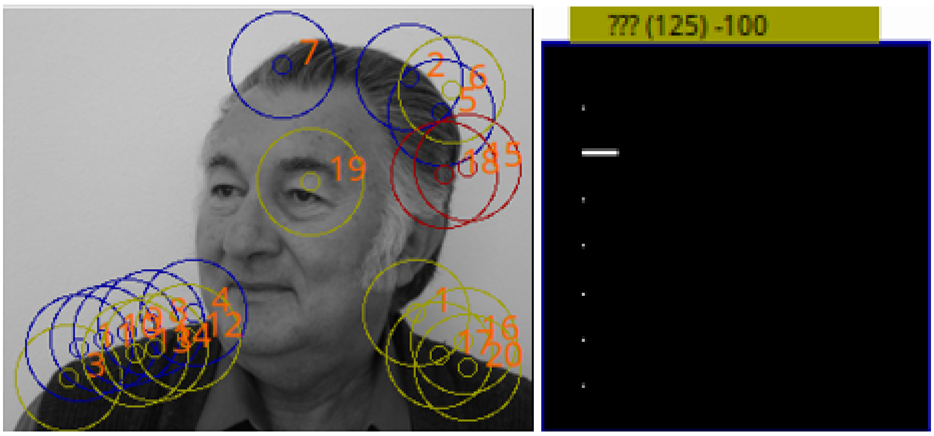
187.png



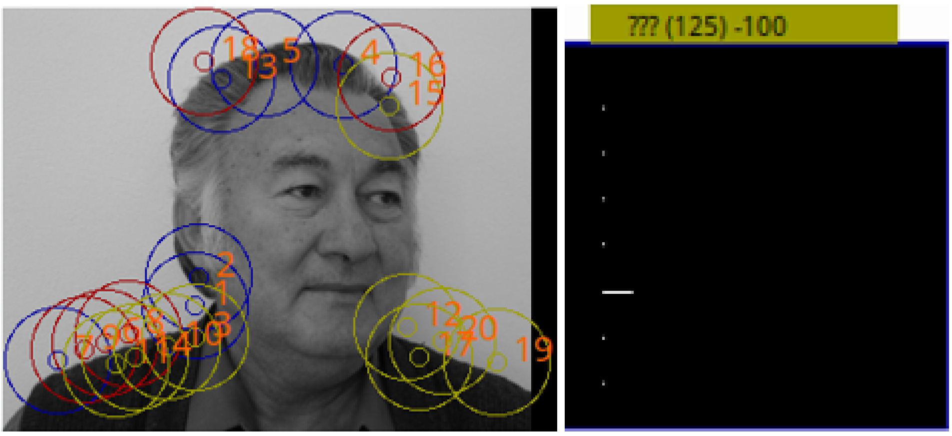
188.png



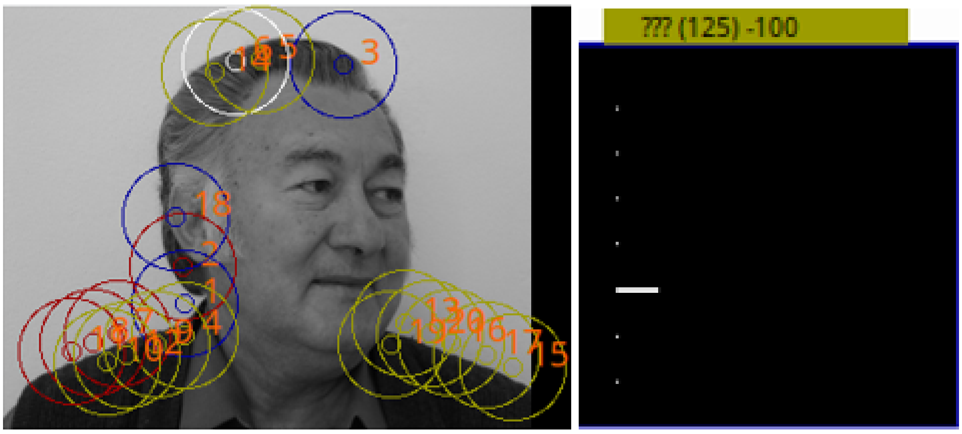
189.png



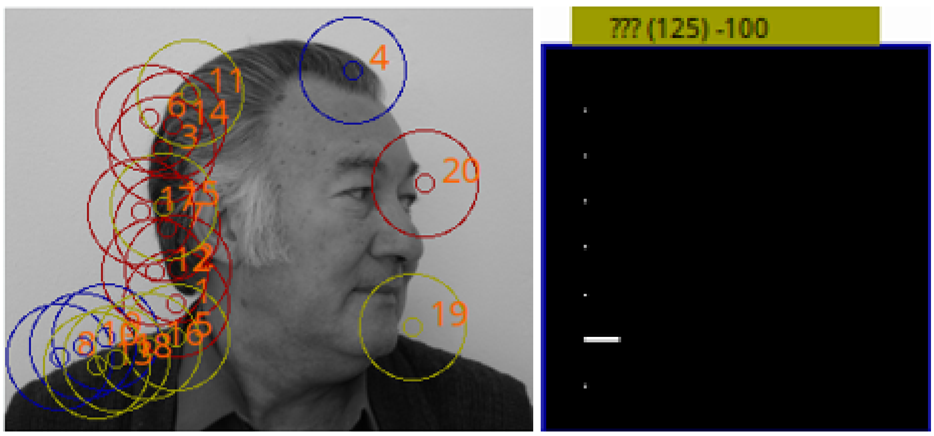
190.png



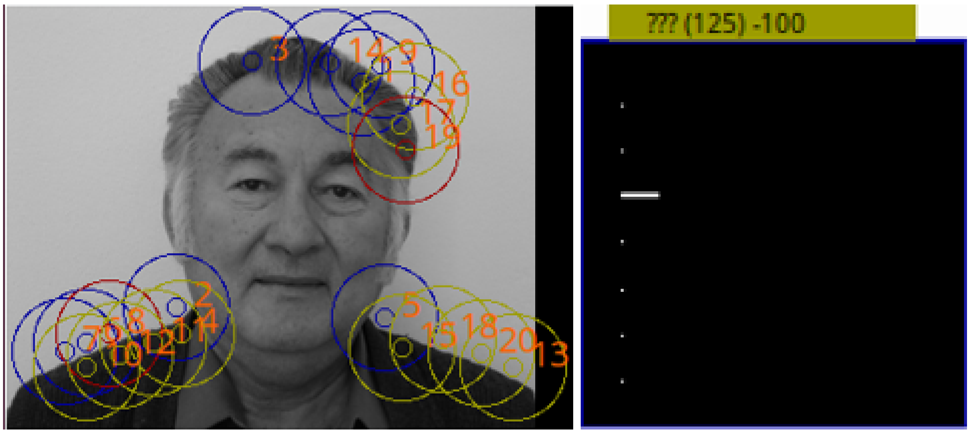
191.png



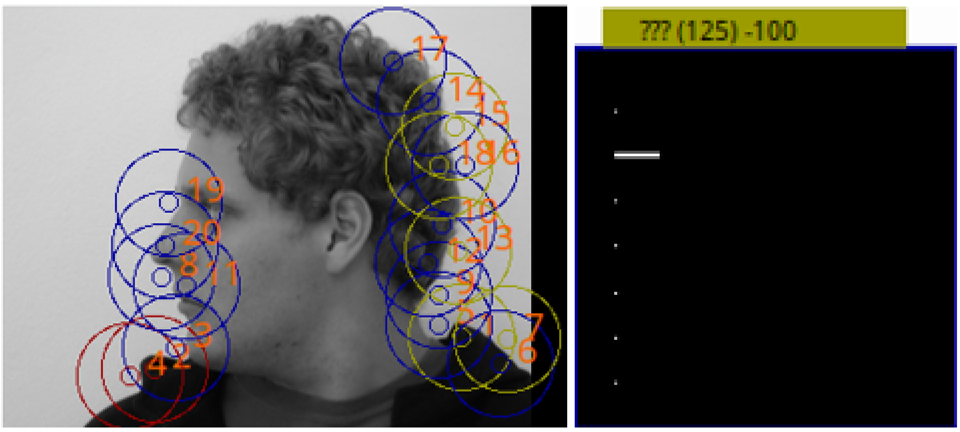
192.png



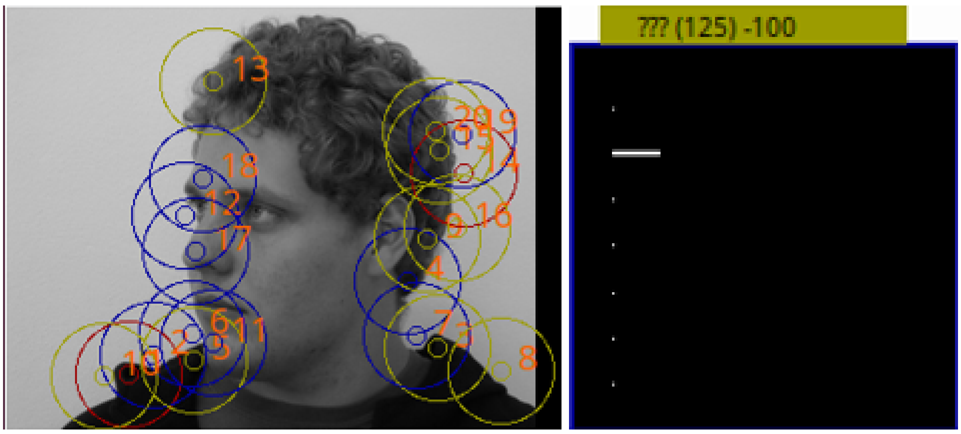
193.png



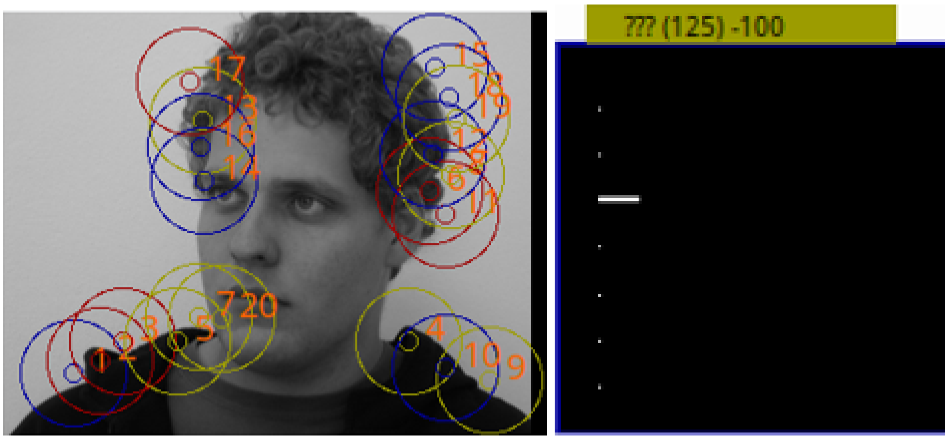
194.png



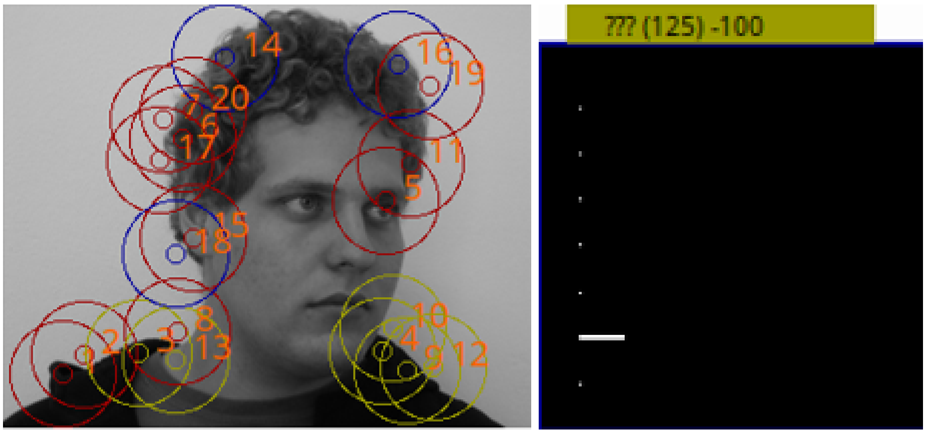
195.png



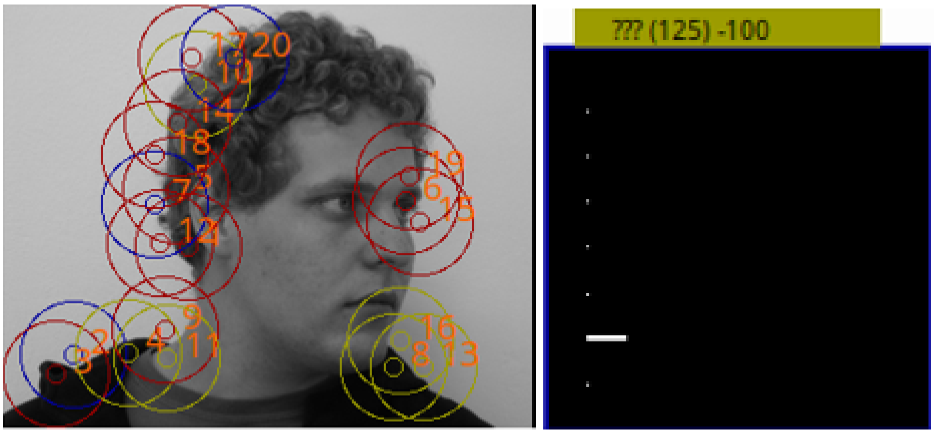
196.png



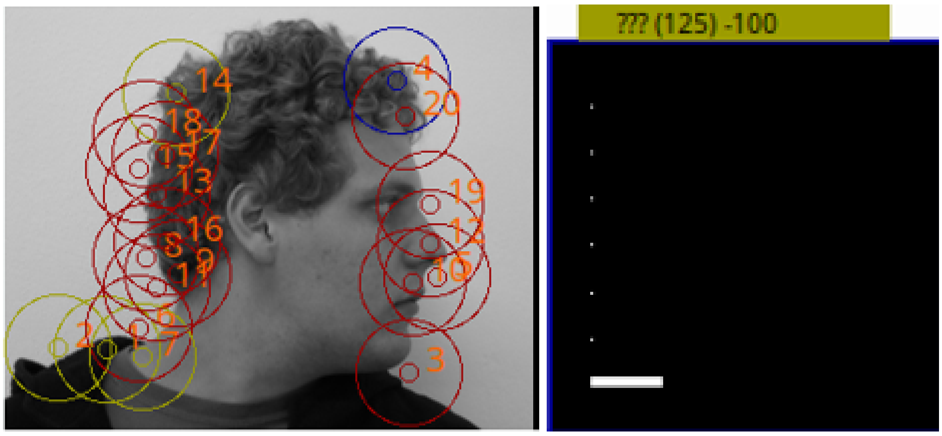
197.png



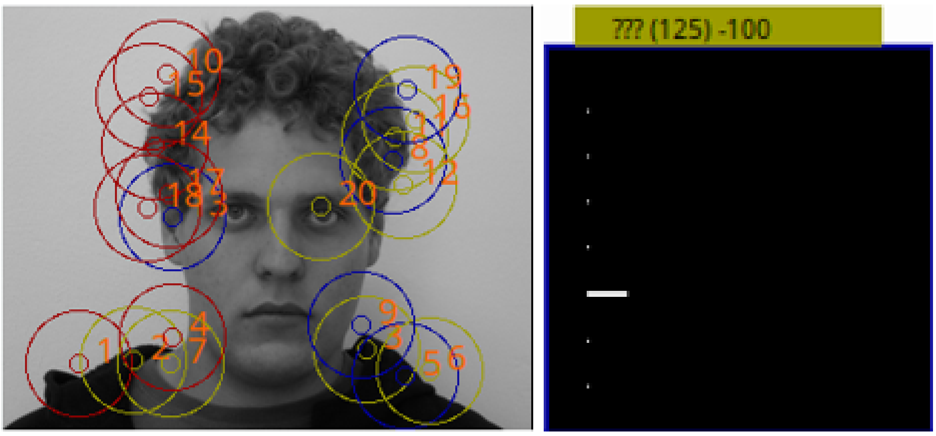
198.png



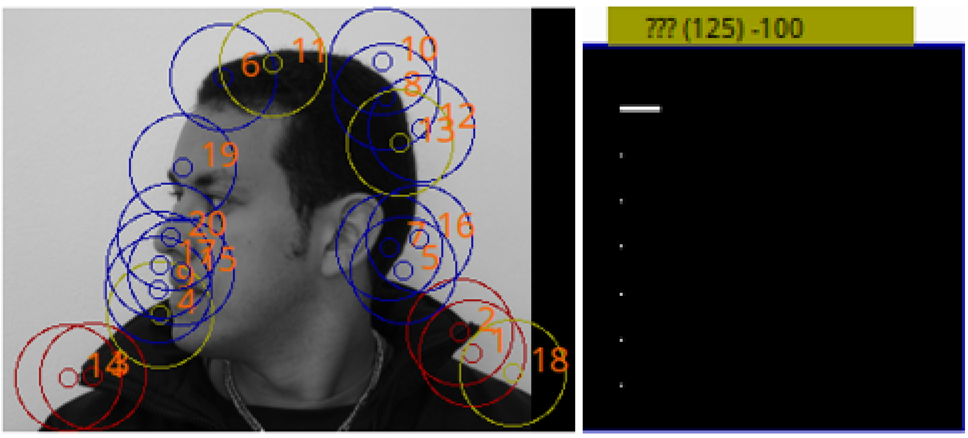
199.png



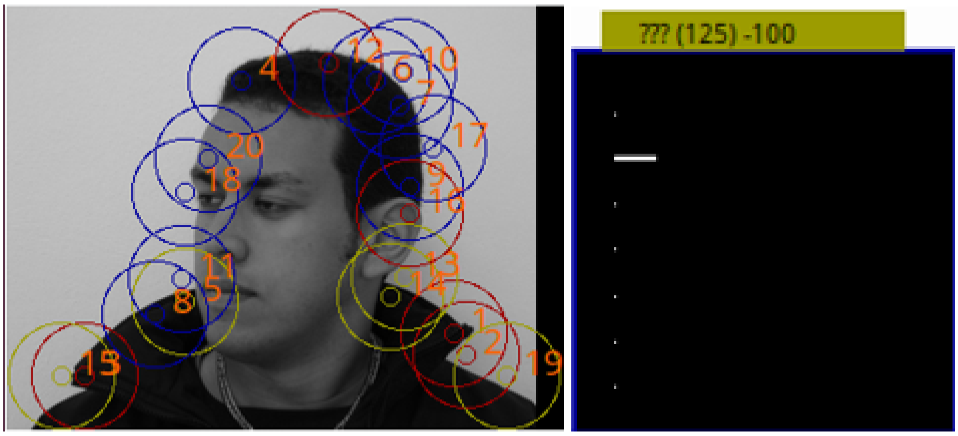
200.png



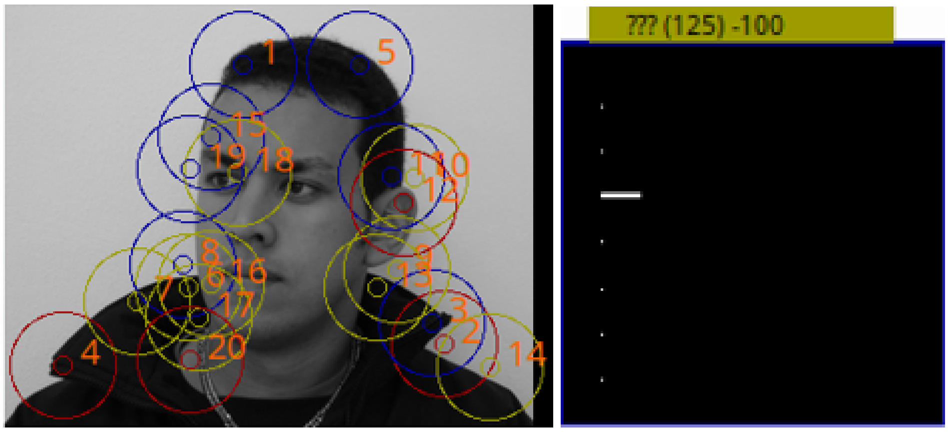
201.png



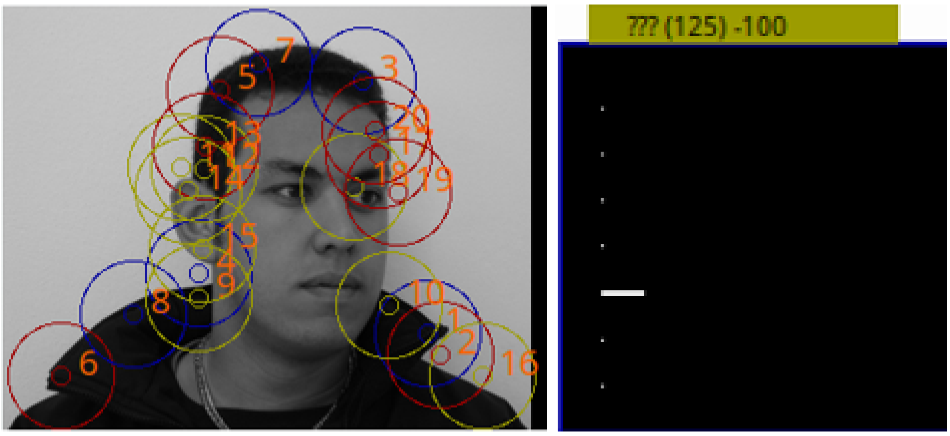
202.png



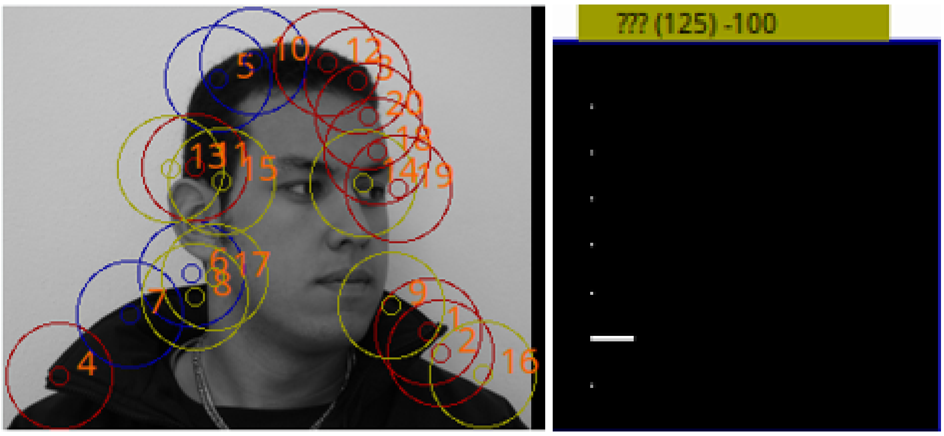
203.png



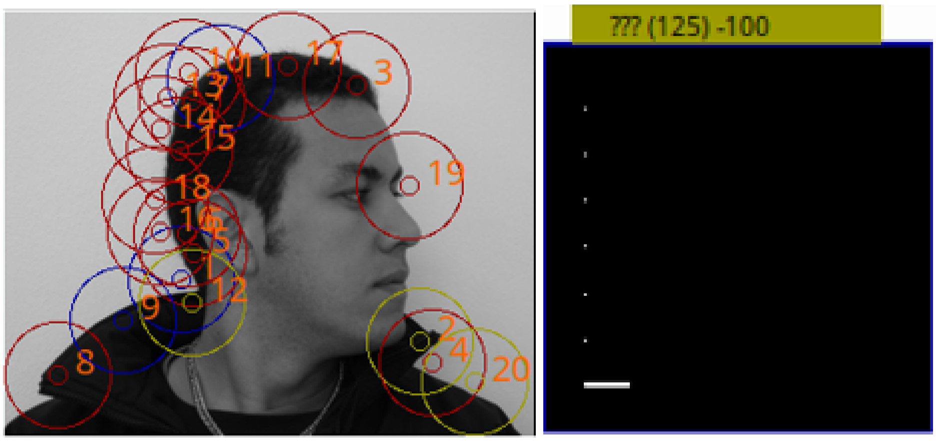
204.png



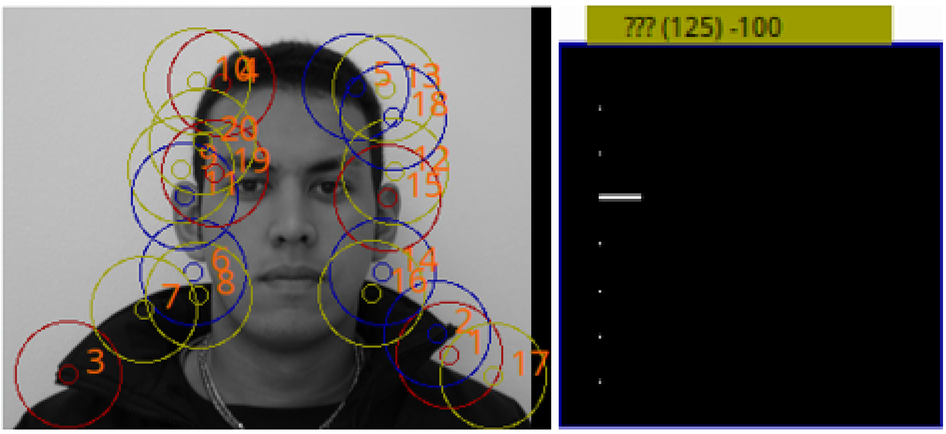
205.png



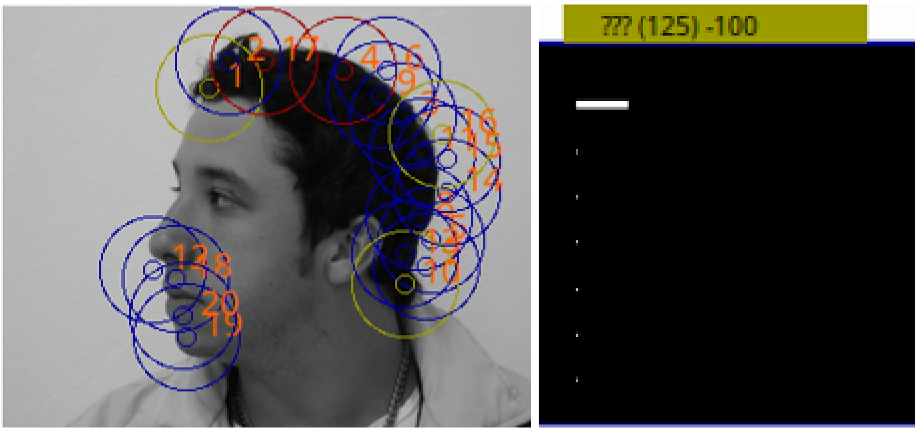
206.png



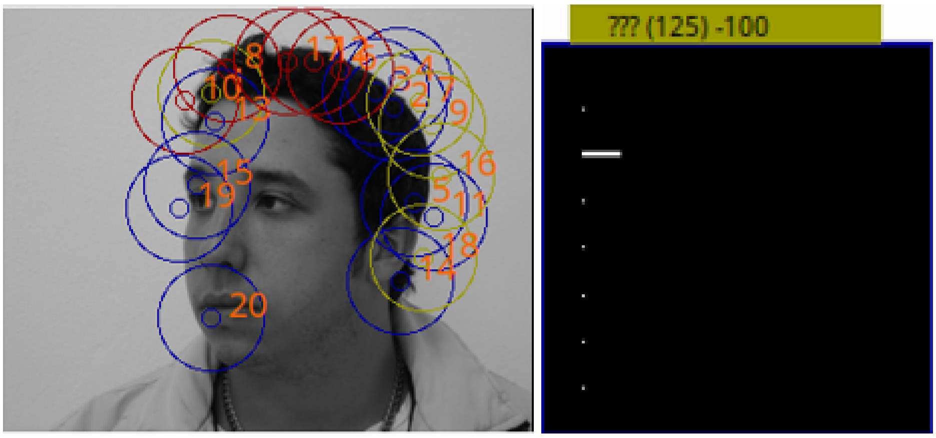
207.png



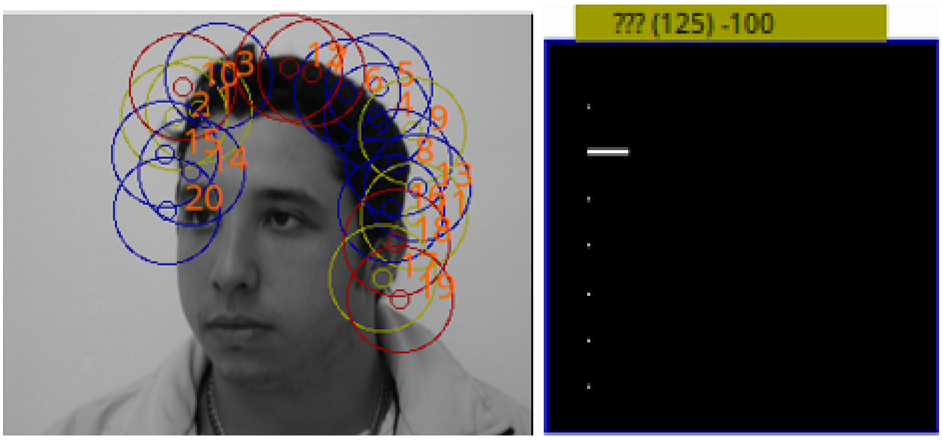
208.png



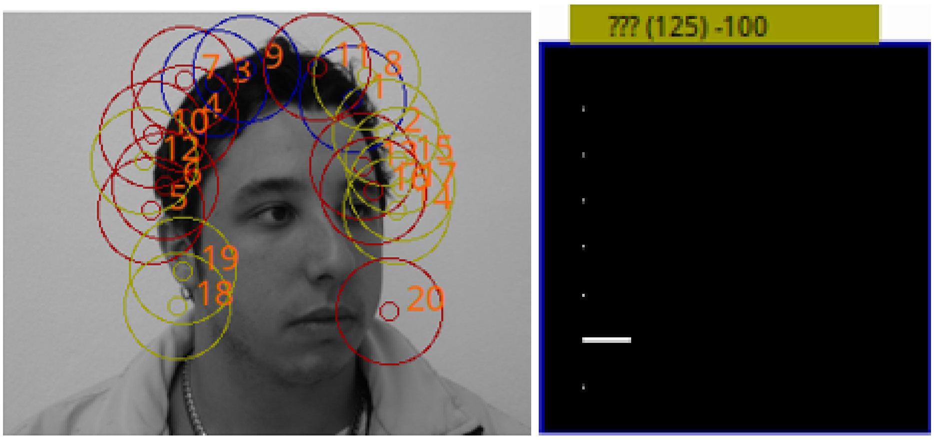
209.png



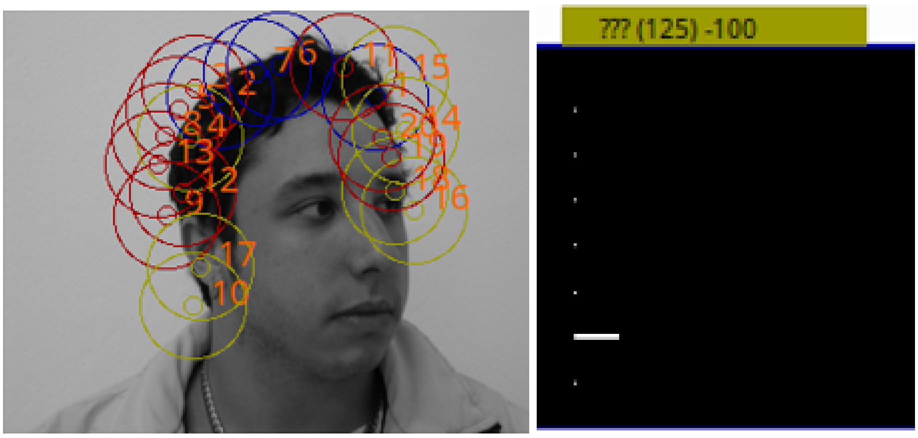
210.png



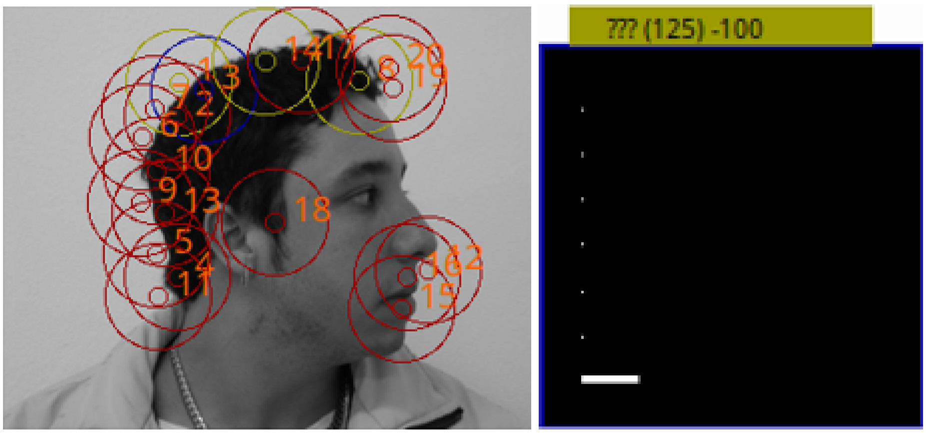
211.png



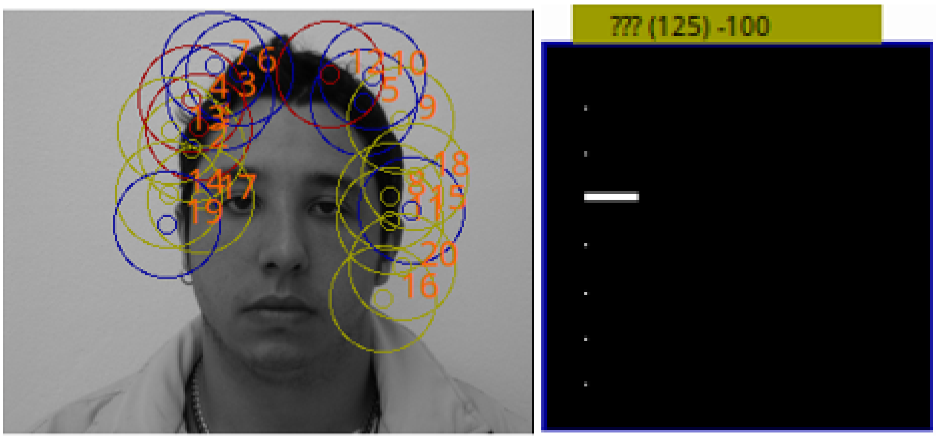
212.png



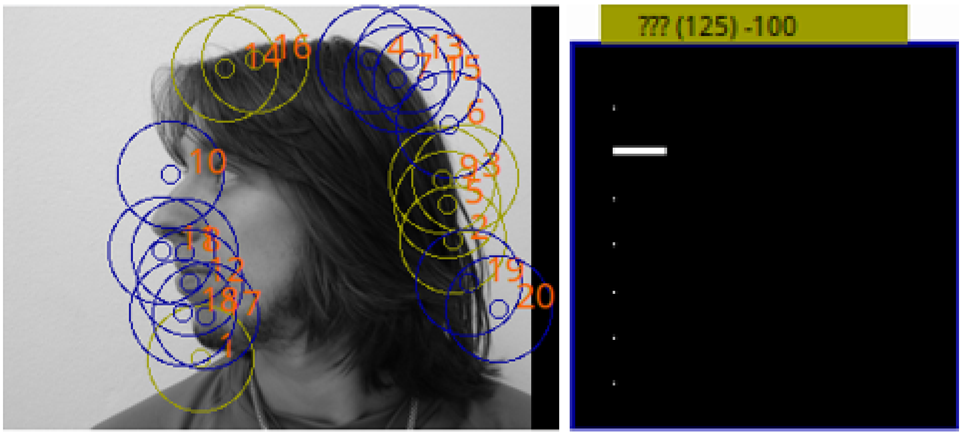
213.png



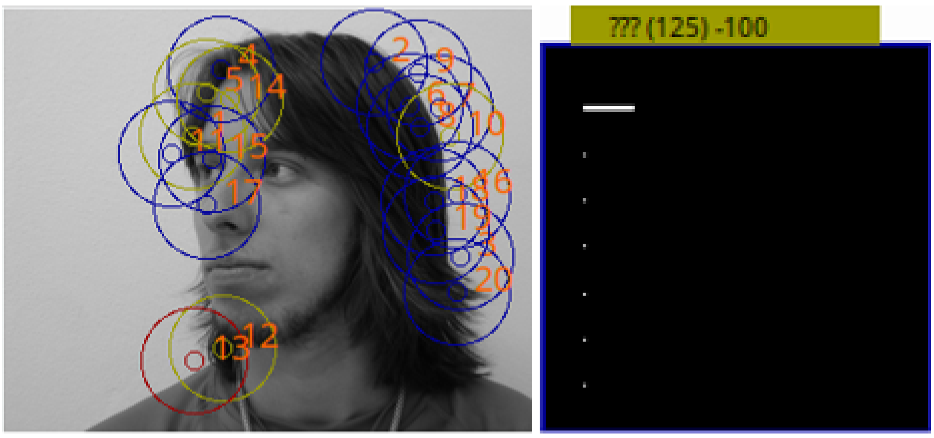
214.png



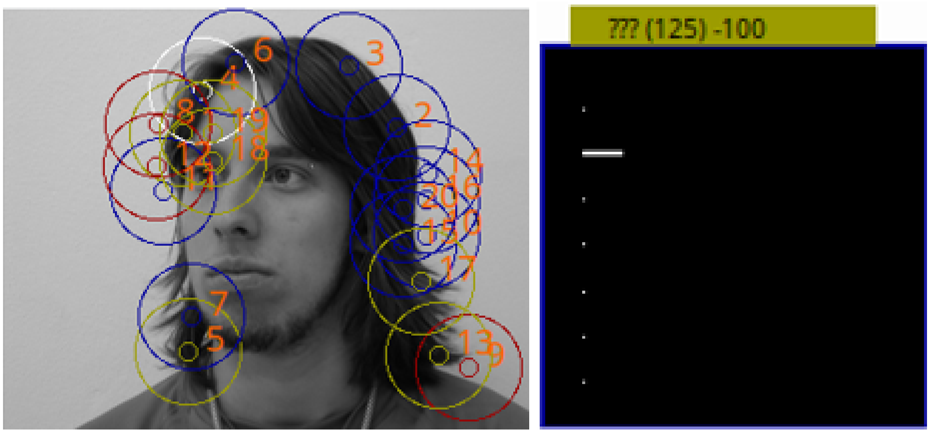
215.png



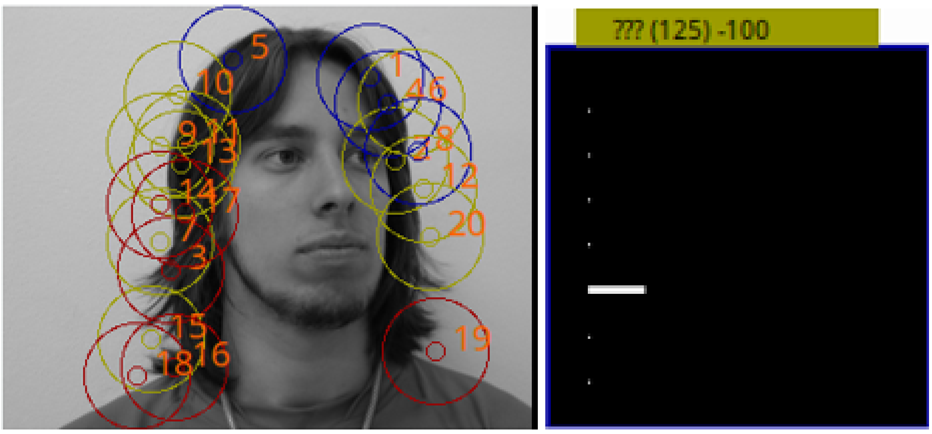
216.png



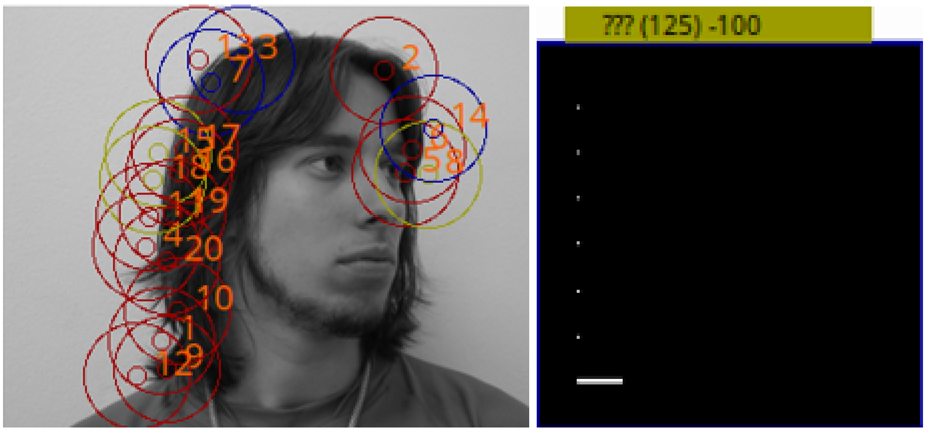
217.png



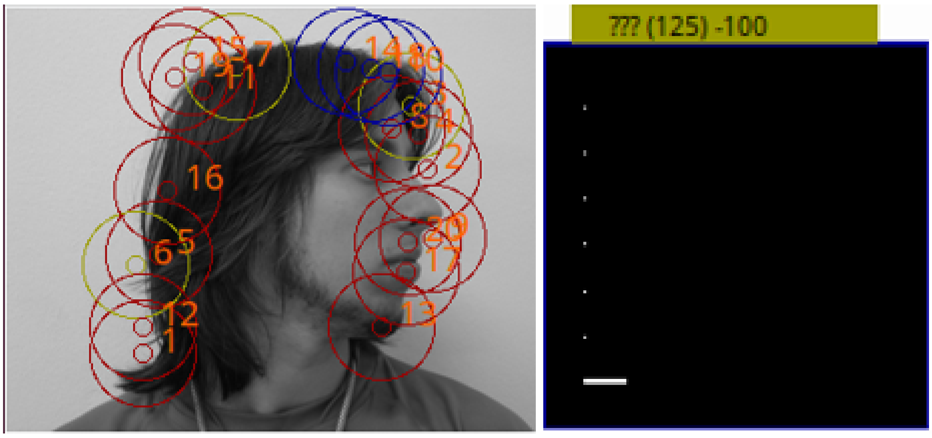
218.png



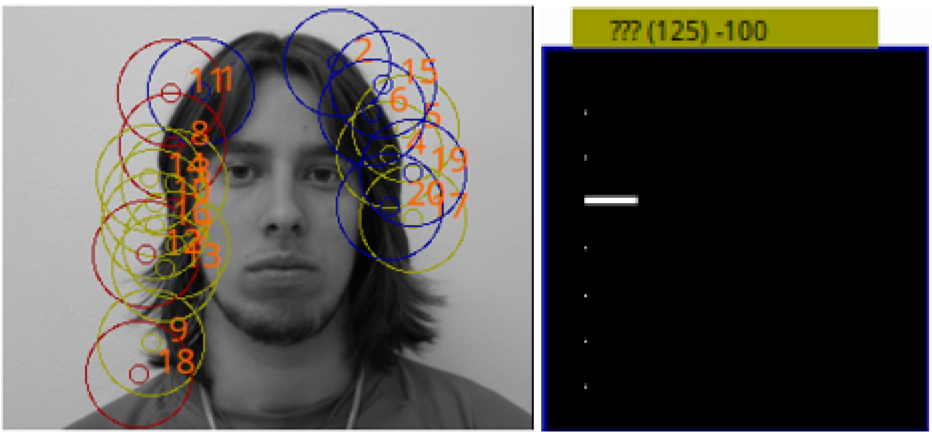
219.png



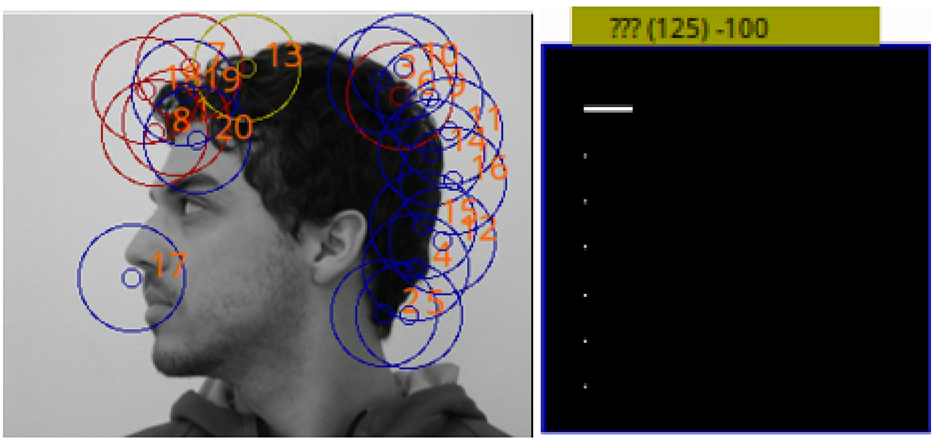
220.png



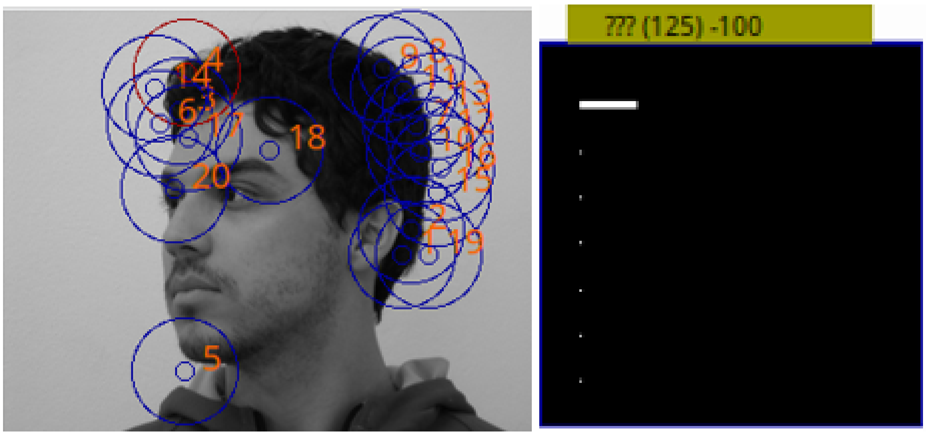
221.png



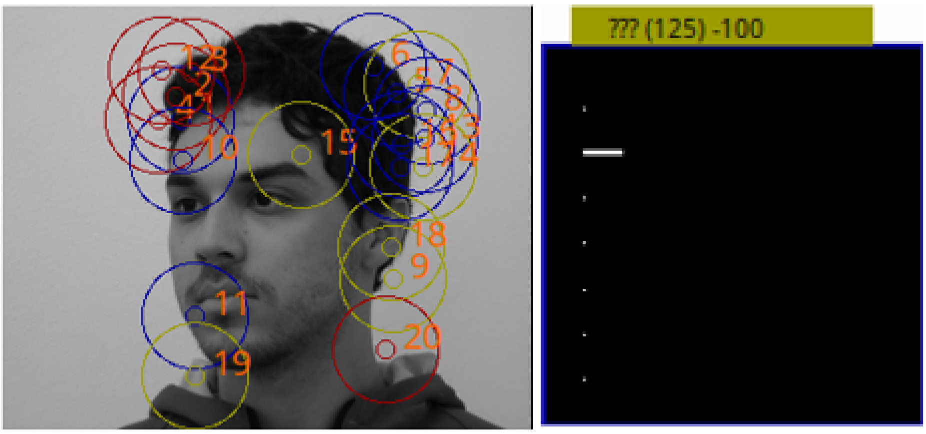
222.png



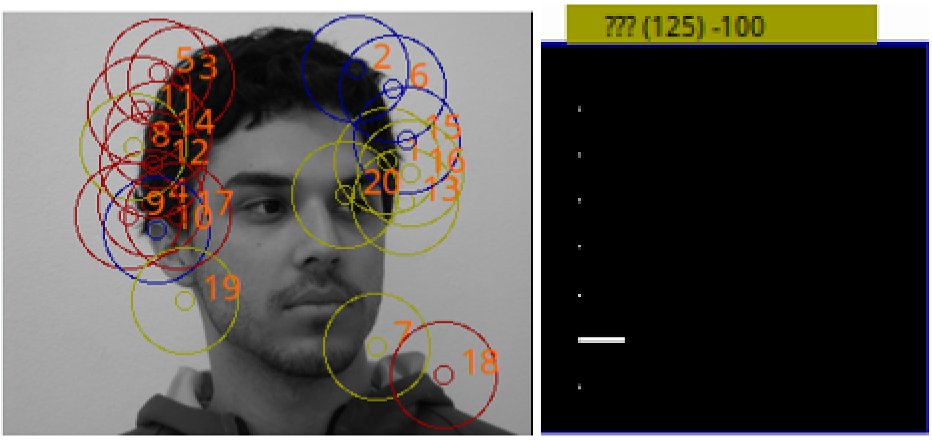
223.png



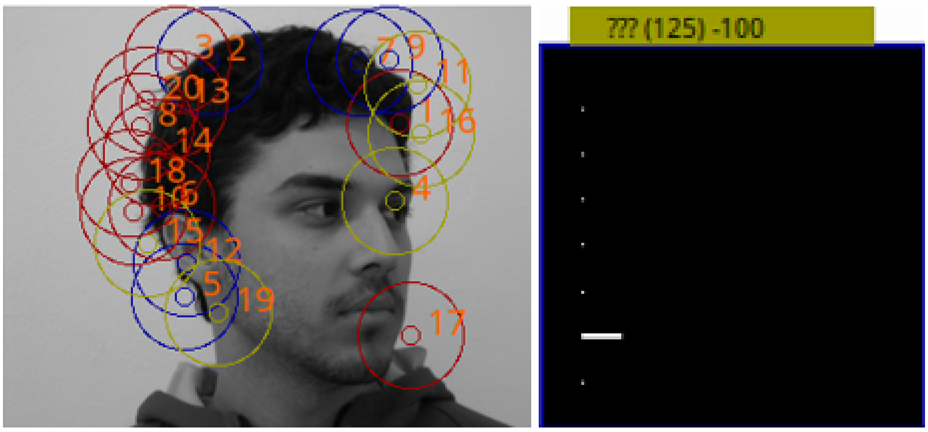
224.png



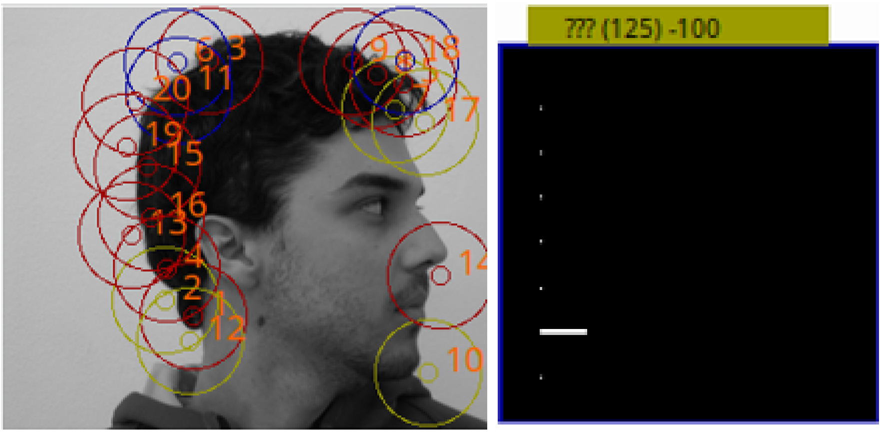
225.png



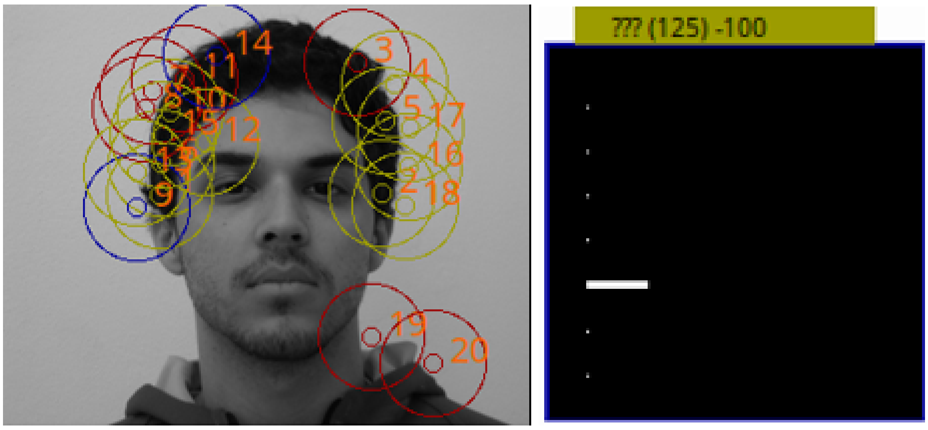
226.png



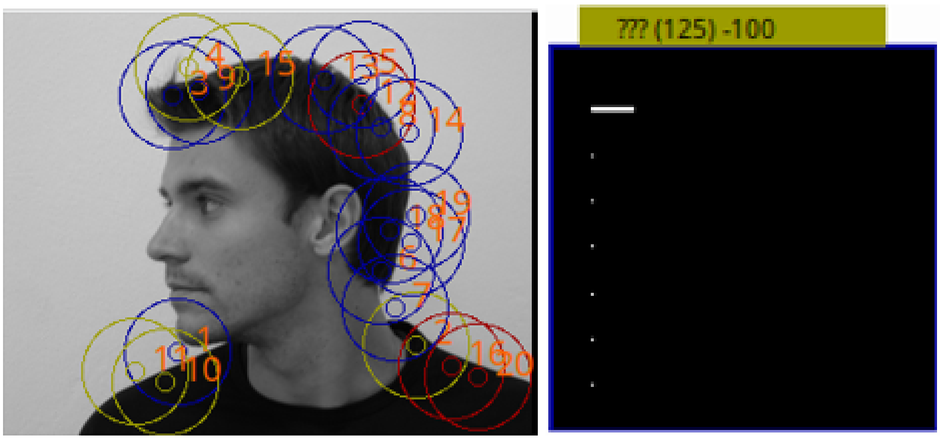
227.png



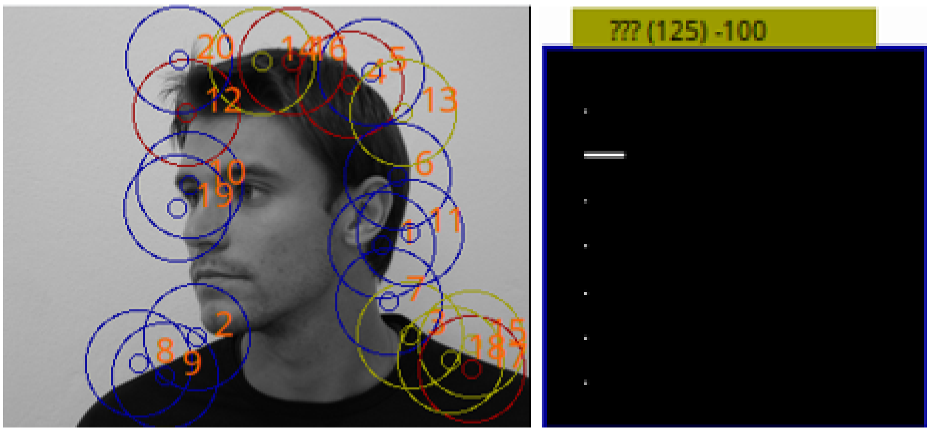
228.png



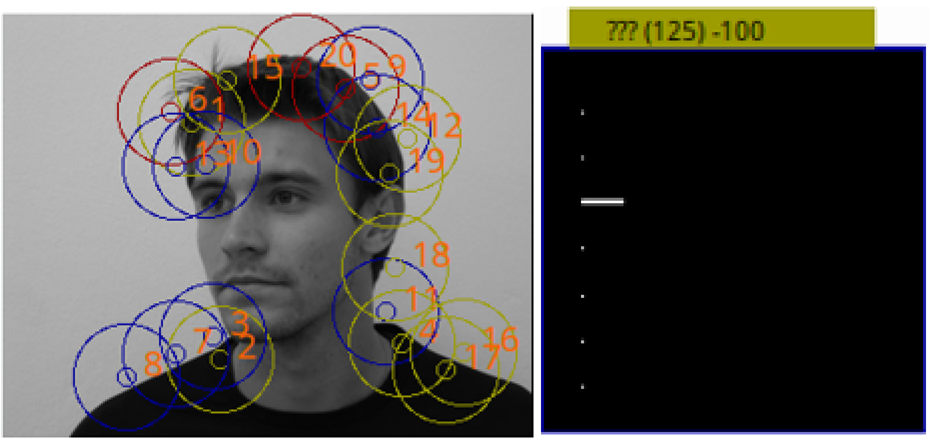
229.png



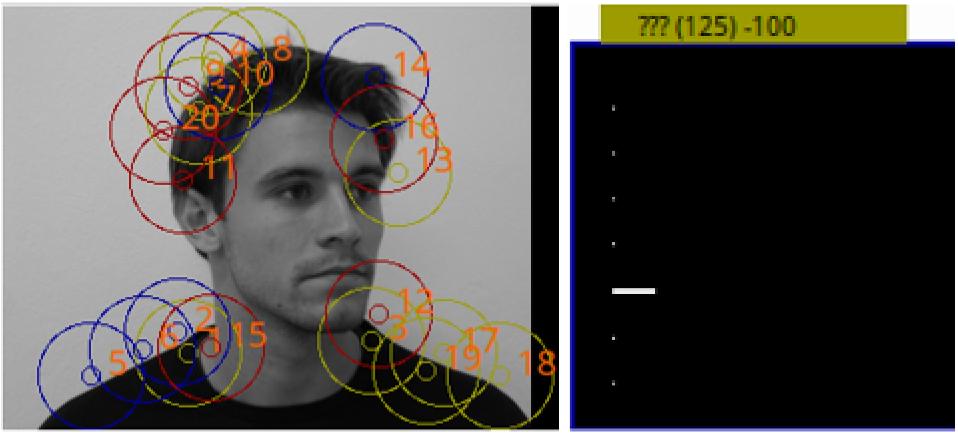
230.png



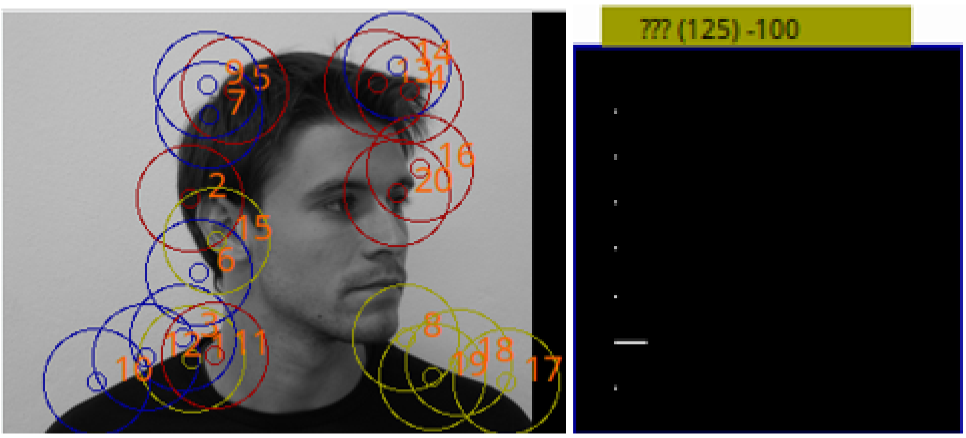
231.png



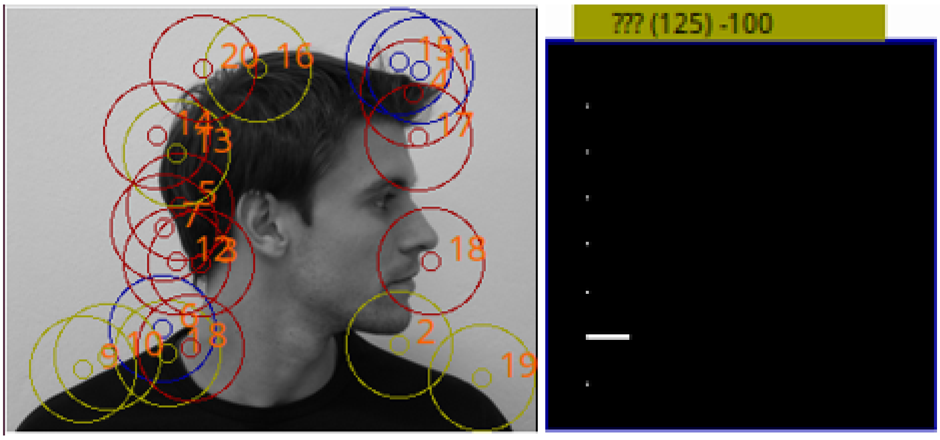
232.png



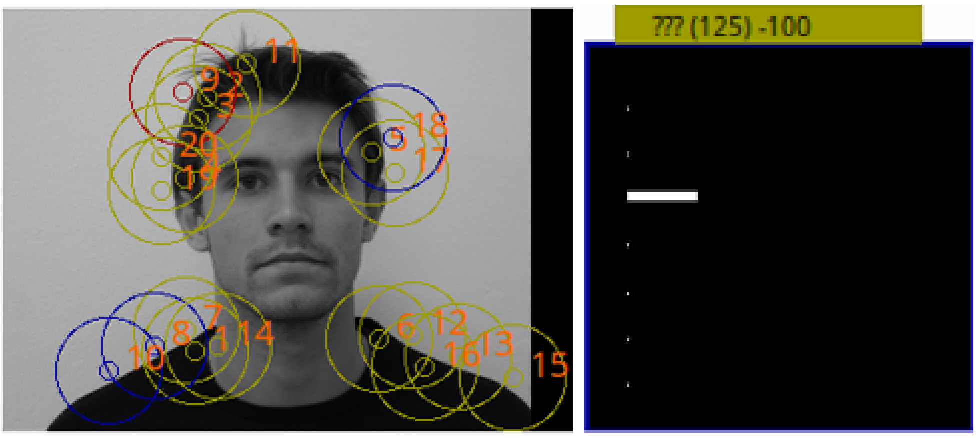
233.png



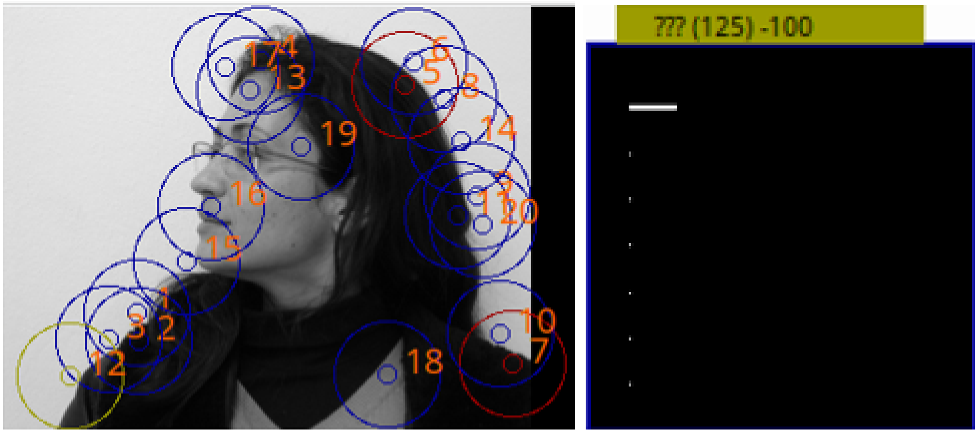
234.png



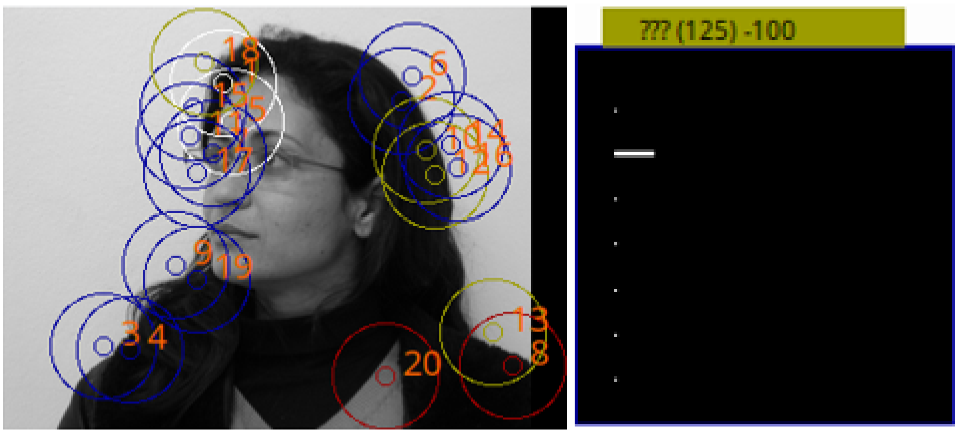
235.png



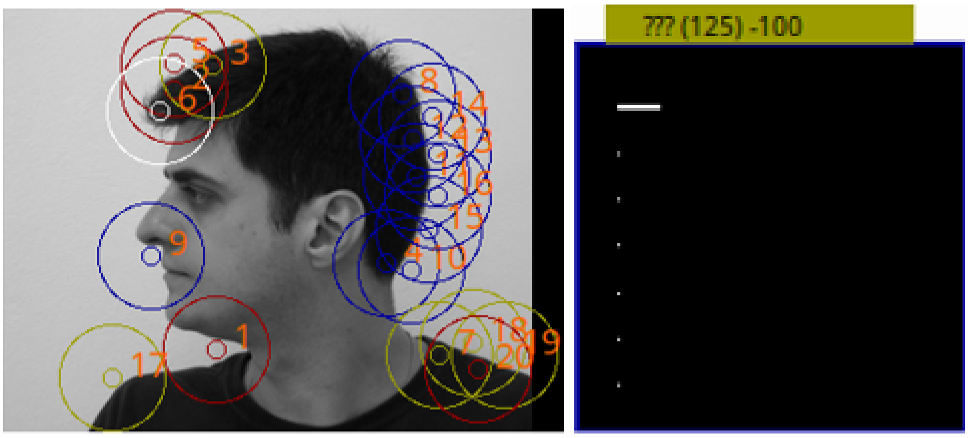
236.png



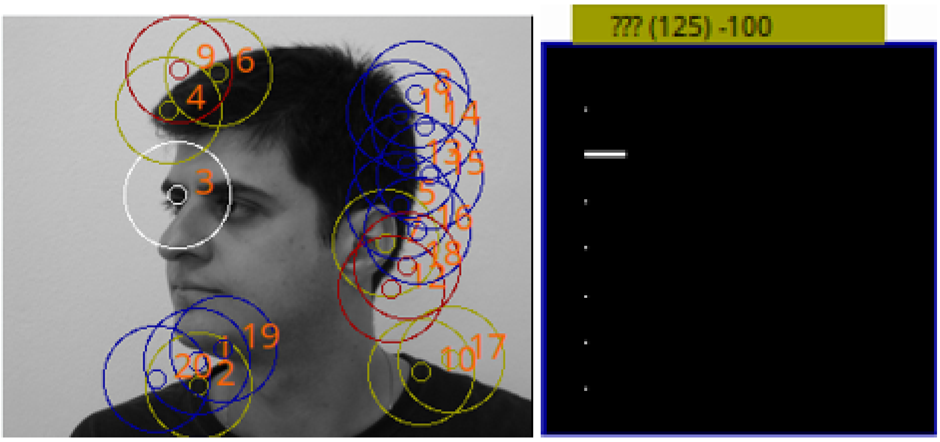
237.png



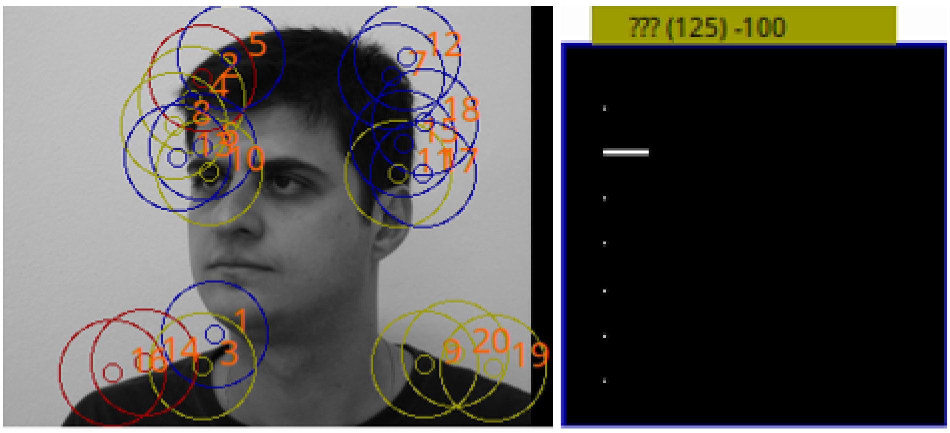
238.png



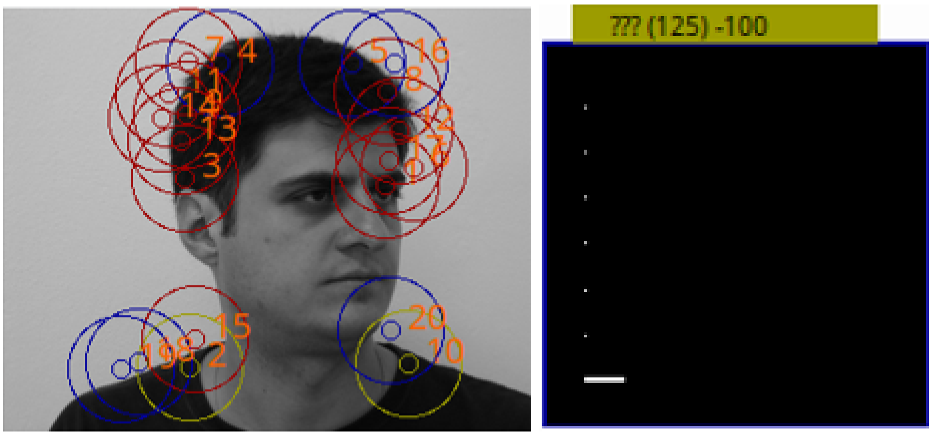
239.png



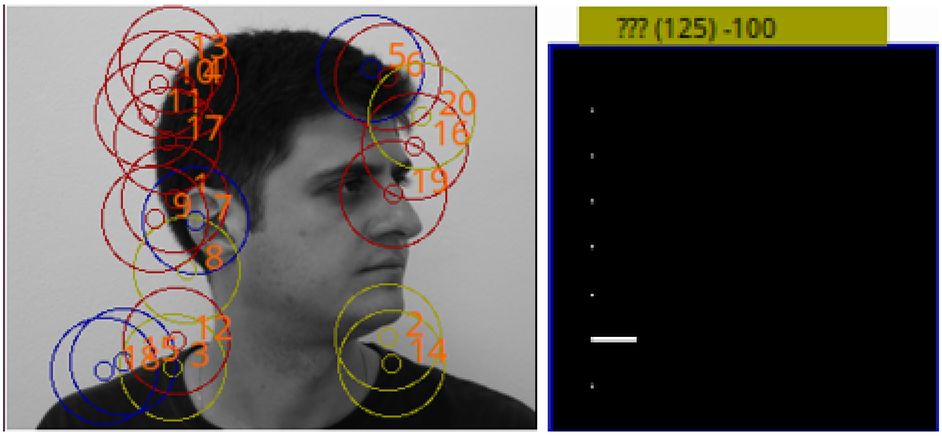
240.png



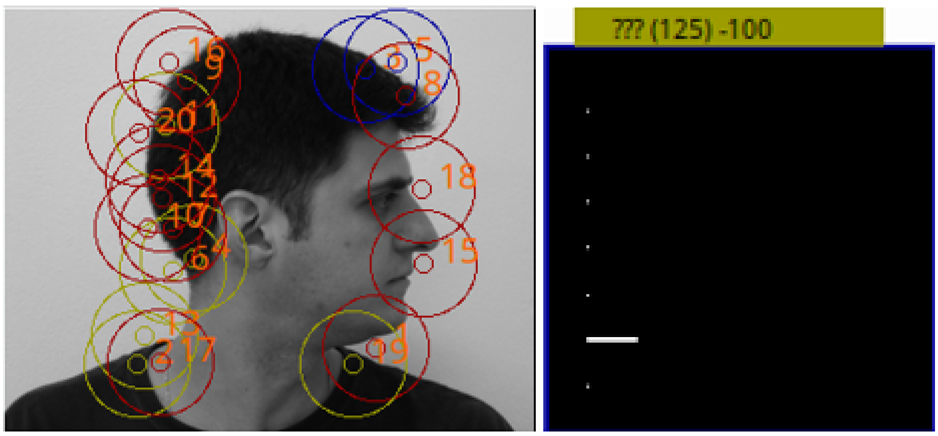
241.png



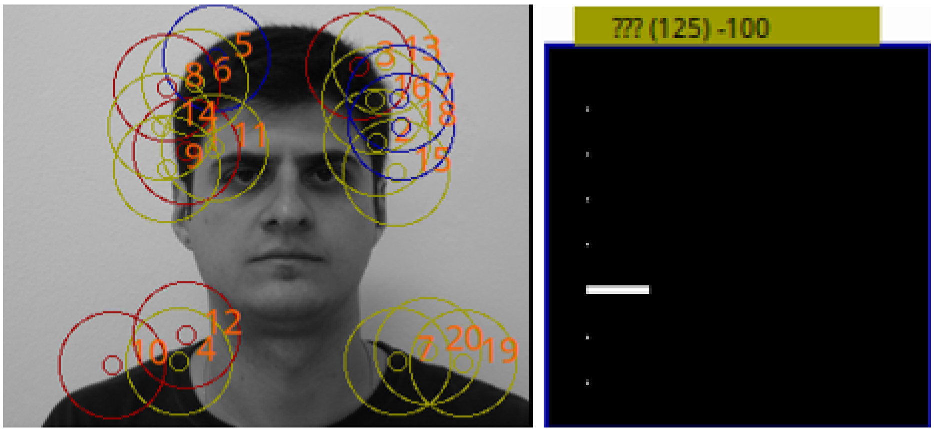
242.png



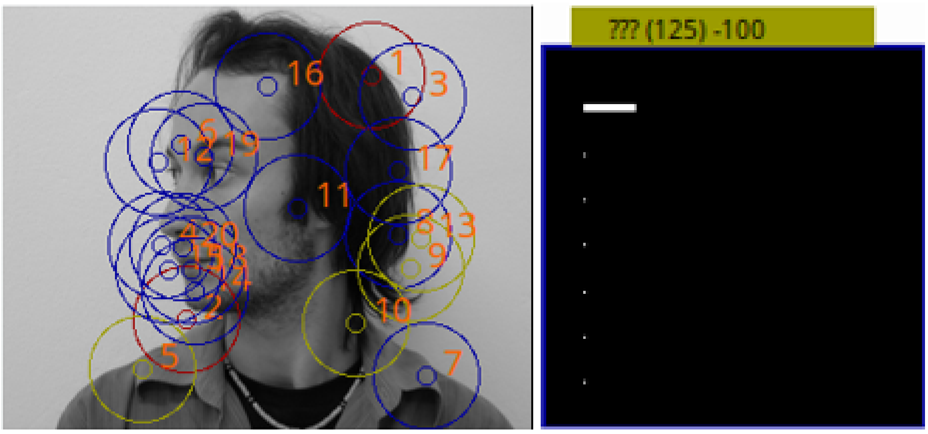
243.png



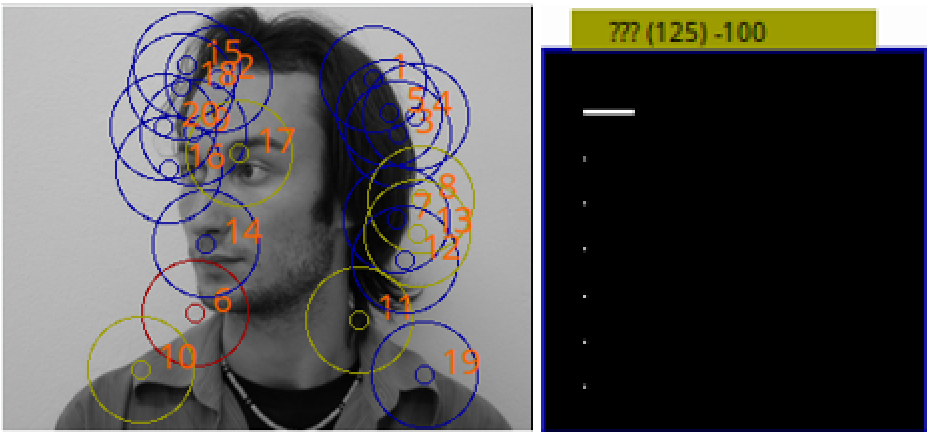
244.png



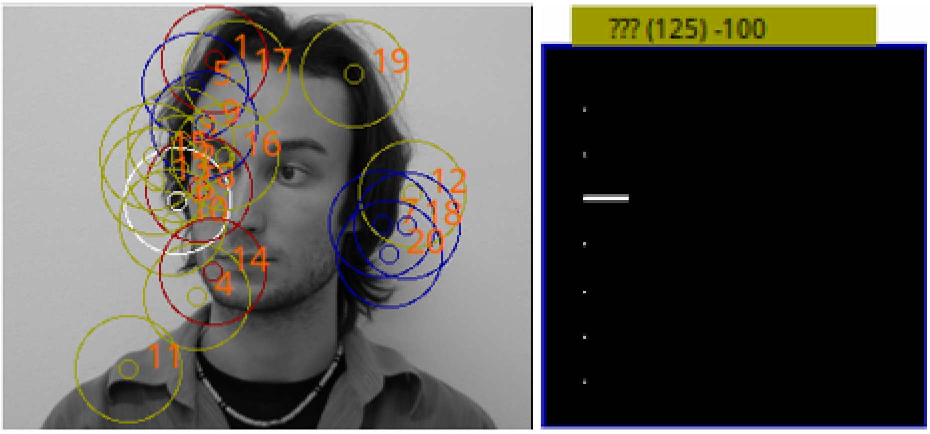
245.png



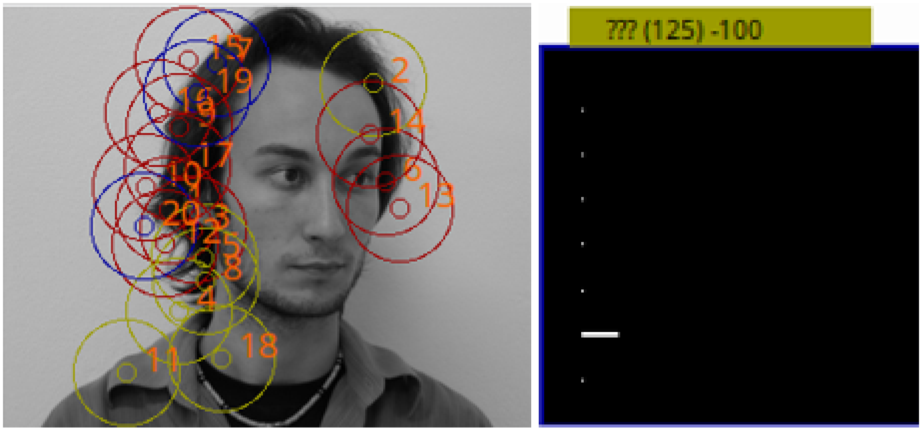
246.png



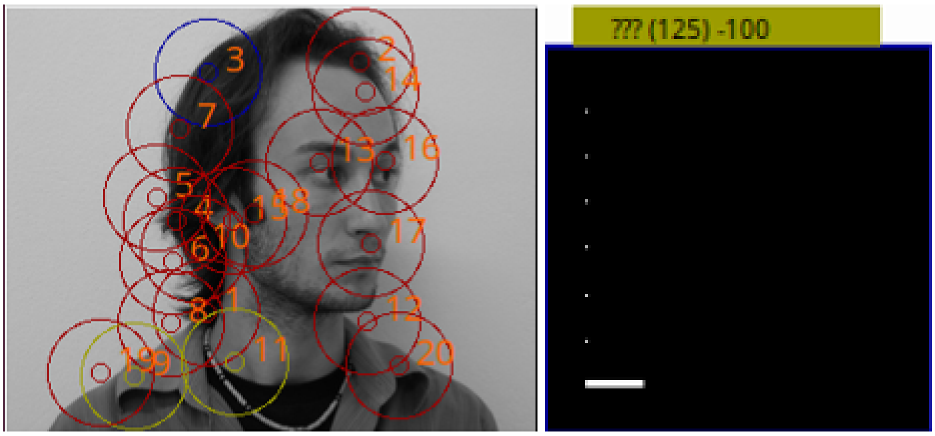
247.png



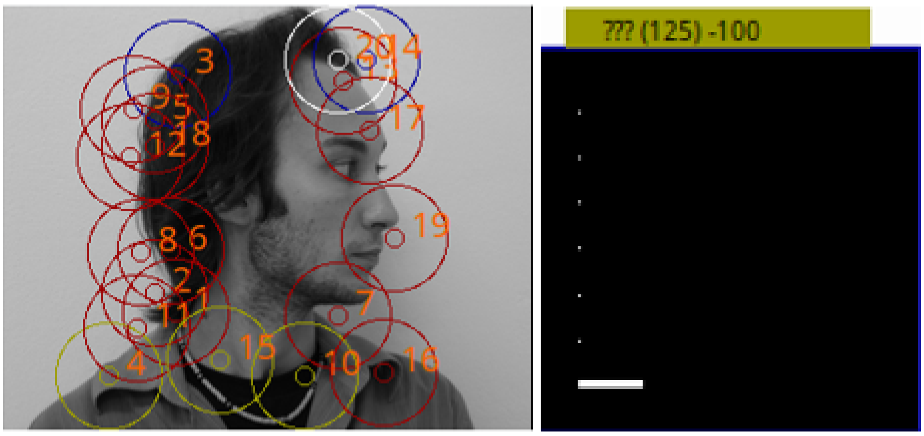
248.png



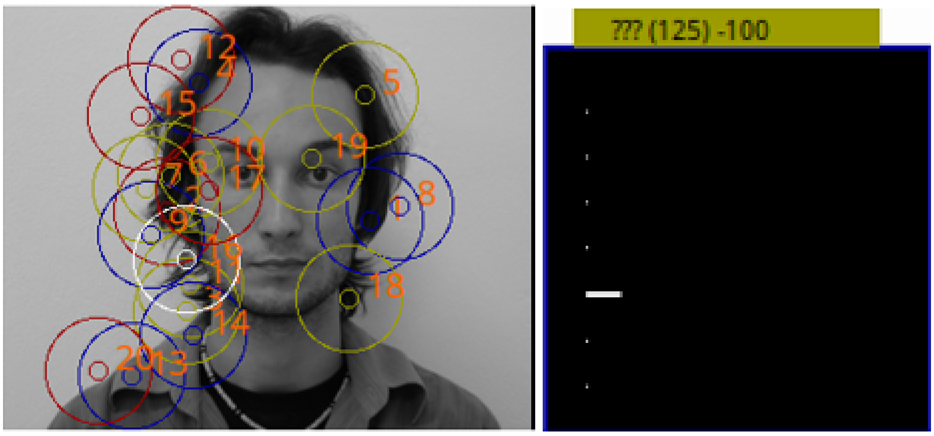
249.png



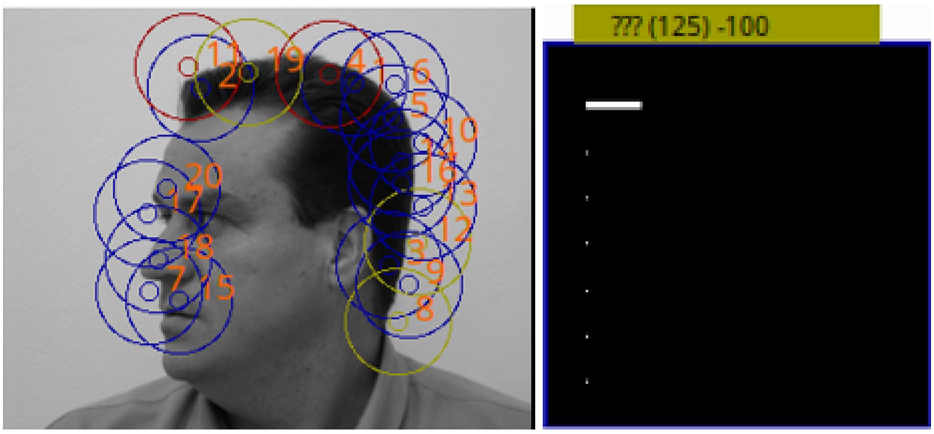
250.png



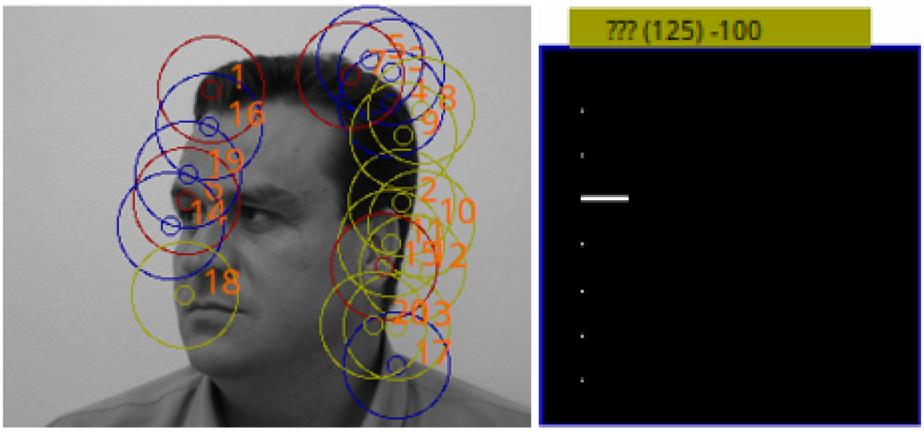
251.png



252.png



253.png



254.png