Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων

Πτυχιακή εργασία

Τίτλος:

A gesture-controlled Serious Game for teaching preschoolers with autism facial emotion recognition

Χριστινάκη Ειρήνη (Α.Μ. : 2456)

Επιβλέπων καθηγητής : Βιδάκης Νικόλαος

Επιτροπή Αξιολόγησης :

Ημερομηνία παρουσίασης:

Acknowledgments

I would like to express the deepest appreciation to my supervisor Dr. Nikolas Vidakis and my sincere acknowledgement in the support and help of my professor Dr. Georgios Triantafyllidis. In addition I would like to express my utmost gratitude to Dr. Manolis Tsiknakis. I am also heartily thankful to my fellow and lab mates in CML and BMI labs.

Abstract

The recognition of facial expressions is important for the perception of emotions. Understanding emotions is essential in human communication and social interaction. Children with autism have been reported to exhibit deficits in the recognition of affective expressions. Their difficulties in understanding and expressing emotions leads to inappropriate behavior derived from their inability to interact adequately with other people. Those deficits seem to be rather permanent in individuals with autism so intervention tools for improving those impairments are desirable. Educational interventions for teaching emotion recognition from facial expressions should occur as early as possible in order to be successful and to have a positive effect. It is claimed that Serious Games can be very effective in the areas of therapy and education for children with autism. However, those computer interventions require considerable skills for interaction. Before age of 6, most children with autism do not have those basic motor skills in order to manipulate a mouse or a keyboard. Our approach takes account of the specific characteristics of preschoolers with autism and their physical inabilities. By creating an educational computer game which provides physical interaction, we hope to support early intervention and to foster learning of emotions.

Keywords: autism, facial emotion recognition, gestures, Kinect, NUI

Table of Contents

List of Figures

List of Tables

List of Acronyms

APA	American Psychiatric Association
API	Application Programming Interface
ASD	Autistic Spectrum Disorder
CDC	Centers for Disease Control and Prevention
CDD	Childhood Disintegrative Disorder
DSM-IV-TR	Diagnostic and Statistical Manual of Mental Disorders Fourth Edition Text Revision
HCI	Human Computer Interaction
ICD-10	International Classification of Diseases 10th Revision
MMWR	Morbidity and Mortality Weekly Report
NIHM	National Institute of Mental Health
NUI	Natural User Interface
PDD	Pervasive Developmental Disorders
PDD-NOS	Pervasive Developmental Disorder - Not Otherwise Specified
RGB	Red Green Blue
SDK	Software Development Kit
USB	Universal Serial Bus
WHO	World Health Organization

1. Introduction

Educating preschoolers with autism and particularly teaching them to recognize and identify emotions from facial expressions is a significant challenge and a complex task. The ability to recognize facial expressions of emotions is crucial to establish interpersonal connections in early life. Although various computer games have been developed to help individuals with autism in emotion recognition tasks, a lack of computer programs specifically written for preschoolers with autism has been observed. A gesture-based Serious Game for facial emotion recognition could be an alternative early intervention proposal.

This project describes an attempt to identify the potential of the new generation game controllers to support learners with Autism Spectrum Disorder (ASD). Microsoft Kinect is an innovative motion-tracking device, designed to provide controller-free interaction through a Natural User Interface (NUI). With Kinect children are able to interact with the game with their body in a natural way. Through the use of gesture-based interaction we expect to help children with autism to improve their skills as early as possible.

1.1 Motivation

Autism is a complex lifelong brain disorder that affects the way the brain uses or transmits information. People with autism see, experience and understand the world in a different way.

The number of children diagnosed on ASDs has risen rapidly. Autism has reached epidemic proportions and prevalence has nearly doubled over the past decade. The median of prevalence estimates of ASD is 62/10.000 (1 in 160).

There is no cure for autism and treatment refers to interventions that aim to help people with ASD to adjust more effectively to their environment. Early intervention can improve the core symptoms of ASD in very young children. Targeting the core social deficits of ASD in early intervention programs is believed to produce sustained improvements in social and communication skills. Appropriate treatments and interventions, environmental support and early diagnosis can improve quality of life for individuals with autism.

Research on autism has attracted considerable scientific interest and scientists have been confronted with the extreme challenge to understand and treat autism. New technologies combined with the current scientific interest in autism could yield new and important information and more effective interventions.

The motivations of this report were the dramatic increase in the incidence of autism, its unique characteristics and the need for interventions specially designed for children with autism.

1.2 Purposes

The main goal of this project is to develop a Serious Game that could assist preschoolers with autism to recognize and understand emotions through facial expressions. This software aims to help them to properly interact with others people, and as a result to increase their inclusion in society.

1.3 Initial problem statement

Autism is characterized by three main impairments: repetitive stereotyped patterns of behaviors, activities, or interests; impairments in communication; and impairments in social interactions. The social interaction deficits involve difficulties in understanding and expressing emotions. Facial features play an important role in emotion recognition. Because the face is the primary canvas used to express distinct emotions nonverbally, the ability to understand facial expressions is particularly vital. Individuals with autism often experience difficulties recognizing and understanding facial expressions of emotions, thus they need support to enhance those skills.

“How can we help individuals with autism to improve social and communication skills?

1.4 Outline

This project starts with an Introduction chapter where the motivation, the purposes and the outline are presented. The Chapter 2 includes the methodology and the method chosen for this project. In the Chapter 3, the problem is defined. The Chapter 4 contains the background research. In the Chapter 5 the problem is re-defined. In the Chapter 6 the requirements are specified. The Chapter 7 describes the design process. The Chapter 8 the implementation section is presented and the Chapter 9 contains the discussion and the future work proposes.

2. Methodology

A program is created through the generation of code and the use of different tools. Software Development Methodology basically refers to the process itself and the way in which it is managed, to ensure that development is completed on time and within a reasonable budget. There are a number of forms that software development methodology can take, depending on the nature of the program being created and the tools used.

The development process selected, as appropriate, for this project is the Engineering Design Process. This process involves a series of steps that lead to the development of a new product or system. It is not a linear process because designers may jump back and forth between the steps as they move toward the final solution.

2.1 Method

The steps of the engineering design process are the above:

1. Define the problem
2. Do background research
3. Re-define problem
4. Specify requirements
5. Create alternative solutions, choose the best one and develop it
6. Build a prototype
7. Test and redesign as necessary
8. Communicate results

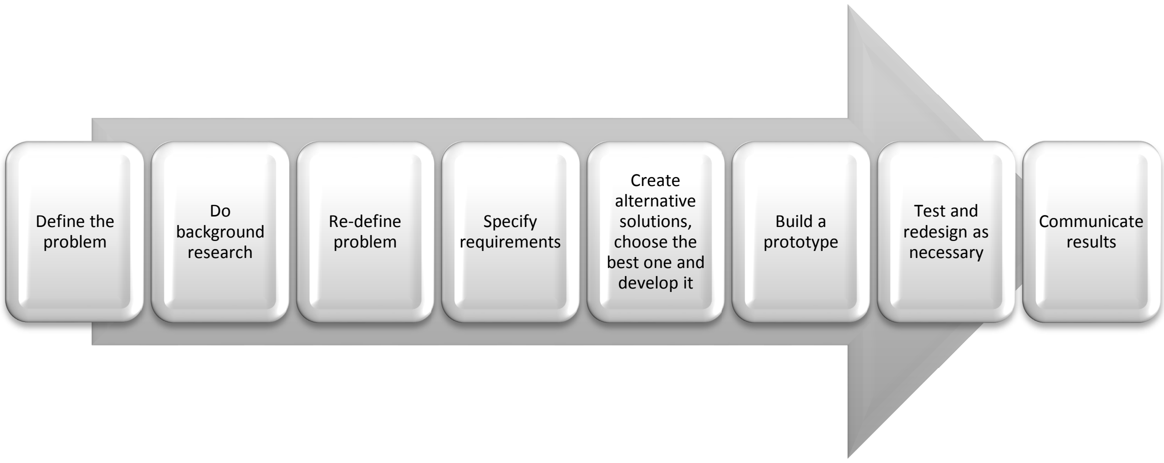


Figure 1: The Engineering Design Process

These were the series of steps that were followed in order to design and implement this project. As a precursor to formulating the project, information was gathered and research conducted to gain an understanding of the needs and challenges that needed to be addressed. A variety of possible solutions were examined. We eventually selected the most promising case, and embarked upon a design that included drawings, analytical decisions on the materials and construction, manufacturing and fabrication technologies to be used. This was preceded by the creation of a prototype and further improvements were made until the product designed was good enough to meet the requirements.

3. Pre-analysis

Autism is a neurodevelopmental condition characterized by social and communicational difficulties alongside circumscribed interests and a strong preference for sameness and repetition. Difficulties understanding emotional state play a major role in social interaction. Face is crucial for the recognition and understanding of emotions and for assisting communications and interactions between people. Training programs focus on teaching emotion recognition, can help individuals with autism to improve their social and communication skills.

3.1 Autism

Autism is a Pervasive Developmental Disorder (PDD) characterized by impairments in social interaction, in communication and by restricted, repetitive and stereotyped patterns of behavior, interests and activities. Children with the same diagnosis of autism may exhibit different symptoms and may demonstrate markedly different behaviors and skills. The spectrum of symptoms can range from mild to severe and efficient diagnosis is difficult. Autism is also described as an ASD due to the variability with which the disorder is manifested. In this project autism and ASD will be referred interchangeably.

The National Institute of Mental Health (NIMH) [1] defines autism as a group of developmental brain disorders, collectively called ASD. The term "spectrum" refers to the wide range of symptoms, skills, and levels of impairment, or disability, that children with ASD can have. ASDs also encompass a wide range of behaviorally defined conditions, which are diagnosed through clinical observation. These conditions have similar symptoms which can vary in severity. Some children are mildly impaired by those symptoms, but others are severely disabled. Autism is just a different operating system and not a processing error.

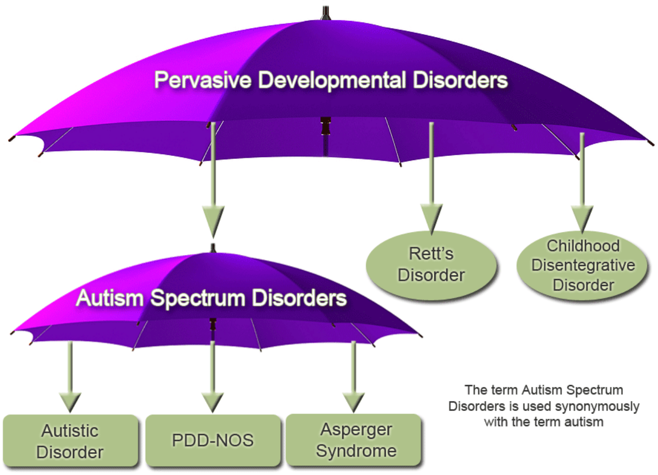


Figure 2: The Autism umbrella

The most authoritative and widely acknowledged definitions of the diagnostic criteria for ASDs are detailed in both the Diagnostic and Statistical Manual of Mental Disorders, Fourth Edition, Text Revision (DSM-IV-TR) [2] and the International Statistical Classification of Disease and Related Health Problems 10th Revision (ICD-10) [3]. The DSM-IV-TR, published by the American Psychiatric Association (APA), presents all categories of mental health disorders and the diagnostic criteria needed to meet them. The ICD-10 is a coding of diseases and signs, symptoms, abnormal findings, complaints, social circumstances and external causes of injury or diseases as classified by the World Health Organization (WHO).

Autism is part of the broader category called Pervasive developmental disorders (F 84) and it is assigned to ICD-10 main categories of Disorders of psychological development (F 80-89). The PDD category includes the syndromes presented in Table 1:

Diagnosis code	Disorder - Syndrome
F84	*Pervasive Developmental Disorders*
F84.0	Childhood autism
F84.1	Atypical autism
F84.2	Rett's syndrome
F84.3	Childhood Disintegrative Disorder
F84.4	Overactive disorder associated with mental retardation and stereotyped movements
F84.5	Asperger syndrome
F84.8	Other pervasive developmental disorders
F84.9	Pervasive developmental disorders, unspecified

Table 1: ICD 10, Pervasive Developmental Disorders

In DSM-IV-TR, autism is a disorder under the category of PDD and is placed within the subclass of Disorders Usually First Diagnosed in Infancy, Childhood or Adolescence. The DSM-IV-TR currently defines five disorders as PDDs presented in Table 2:

Diagnosis code	Disorder - Syndrome
	Pervasive Developmental Disorders
299.00	Autistic disorder
299.80	Rett's disorder
299.10	Childhood Disintegrative Disorder
299.80	Asperger's disorder
299.80	Pervasive Developmental Disorder not otherwise specified (PDD-NOS)

Table 2: DSM-IV-TR, Pervasive Developmental Disorders

Both the APA and the WHO are currently working on revisions of their classification systems. The DSM-5 will be released in May 2013 and the ICD-11 will be released in 2015.

3.1.1 History of autism

The word "autism" (αυτισμός) derives from the Greek word "eautos" (εαυτός) which means self and the suffix -ismos (-ισμός) that indicates a behavior with specific characteristics. It is a neurodevelopmental disorder characterized by impairments in social interaction, in communication and by restricted, repetitive and stereotyped patterns of behavior, interests and activities. It can make the individuals appear self-centered and struggling to see things from other people's perspective.

A greater preponderance of PDD diagnosis in male approximating 4:1 has been consistently reported in the scientific literature. Recent investigations have indicated a potential change to traditional estimates of gender ratios and have reported greater gender disparity [4].

In 1908 Theodor Heller used the name dementia infantilis to describe a disorder that is closely related to the current ASD, called Childhood Disintegrative Disorder (CDD). He reported six cases of children who exhibited severe developmental regression at 3 to 4 years of age after a normal, or near normal, period of mental development. He introduced the condition as dementia infantilis to describe the sudden loss of language and social skills by young children [5]. Without using the word autism, Heller actually seemed to be describing the core 'triad' of impairments that make up the diagnostic criteria for autism.

Autism was first described in 1943 by Leo Kanner, a child psychiatrist at Johns Hopkins University in Baltimore, Maryland. He wrote about eleven children (8 boys and 3 girls) who shared "essential common characteristics" but whose pattern of difficulties were quite different from children diagnosed with other conditions [6]. He observed that the children became upset when other changes to routines occurred and often engaged in repetitive behavior, such as hand-flapping. He had identified a previously undiscovered syndrome and described the outstanding, "pathognomonic[1]", fundamental disorder as the children's inability to relate themselves in the ordinary way to people and situations from the beginning of life. Kanner used the term "autism" and called the disorder "early infantile autism".

A year later, Hans Asperger published a similar series of cases. He essentially made same discoveries at the same time, independently of Kanner, but the patients he identified had no history of speech delay. He also used the term "autistic" and he described similar characteristics of impaired communication and social interaction [7], [8], [9].

3.1.2 Causes of autism

The cause of autism is not yet identified and no one really knows its etiology. However, many scientists believe that several factors combine in the causes of autism. Overwhelming evidence suggests that biological and environmental factors are the possible cause or causes of this disorder.

Medications have been identified as potential autism risk factors when given during pregnancy. Valproic acid, an anti-epileptic drug that is also used for bipolar disorder and schizophrenia is related to ASD [10], [11].

Studies of infants with an older biological sibling with ASD revealed that a total of 18.7% of the infants developed ASD. The same study indicated a twofold increase in the probability of an ASD diagnosis at outcome for infant siblings who had multiple older affected siblings relative to those who had only one older affected sibling [12].

ASD commonly co-occurs with other developmental, psychiatric, neurologic, chromosomal, and genetic diagnoses. The co-occurrence of one or more non-ASD developmental diagnoses is 83%. The co-occurrence of one or more psychiatric diagnoses is 10% [13].

Children born to older parents are at a higher risk for ASDs [14] and a small percentage of children who are born prematurely or with low birth weight are at greater risk for having ASDs [15].

There are also suggestions that the vaccine for measles, mumps, and rubella (MMR) cause autism but there is no scientific evidence to support those claims.

3.1.3 Prevalence of autism

Since 2002 autism has increased dramatically. The United States Centers for Disease Control (CDC) releases autism prevalence estimates as part of their MMWR (Morbidity and Mortality Weekly Report). In the latest MMWR on autism, ASD prevalence estimates in the overall population increased 23% for the 2-year period 2006–2008, and 78% during the 6-year period 2002–2008 [16]. The median of prevalence estimates of ASD is 62/10.000 [17].

Studies in Asia, Europe, and North America have identified individuals with an ASD with an average prevalence of about 1%. A recent study in South Korea reported a prevalence of 2.6% [18].

In Greece systematic epidemiological data do not exist. Autism prevalence is an issue that requires further study, as no studies have been conducted using globally standardized methods and diagnostic tools. On the basis of conservative estimates from other countries, there should be about 15.000 to 20.0000 people with autism in Greece, while the number of children with autism up to 14 years of age is estimated 4.000 to 5.000 [19]. Only one study has been conducted in order to describe the developmental, familiar and educational characteristics of a sample of children with ASD in Greece [20].

3.1.4 Early detection and diagnosis

Recognizing the early signs can be a challenge for parents and healthcare professionals. It is desirable to screen as early as can be done reliably, in order to maximize intervention opportunities. However, screening that is conducted too early may not be able to distinguish ASD from other forms

[1] Specifically distinctive or characteristic of a disease or pathological condition; denoting a sign or other indicant on which a diagnosis can be made.

of developmental delay, or even from typical development. A number of screening and diagnostic tools have been developed in this field. Autism may be screened around 18 months of age and a diagnosis reliably be made around age 30 months. The American Academy of Pediatrics published a policy statement for the identification and evaluation of children with ASD [21]. They suggest screening all children for signs of autism at 18 and 24 months of age, whether or not a parent has any concerns. Validated screen tools such as the CHAT, the CAST and the M-CHAT [22] are chosen because of their easy accessibility and reliability based on current available research.

When considering the diagnosis of ASD, physicians refer to the DSM-IV-TR to determine whether or not a child meets the criteria-based diagnosis for a disorder within the spectrum. The manual outlines twelve possible ‘‘symptoms’’ of ASD within three areas of impairment, i.e. social interaction, communication and restricted and repetitive interests, activities and behaviors as presented in Table 3. These criteria were established for children 3 years and older and may be difficult to apply to younger children.

299.00 Autistic Disorder			
A	Six or more items from (1), (2), and (3), with at least two from (1), and one each from (2) and (3):		
	(1)	Qualitative impairment in social interaction, as manifested by at least two of the following:	
		a	Marked impairment in the use of multiple nonverbal behaviors such as eye-to-eye gaze, facial expression, body postures, and gestures to regulate social interaction.
		b	Failure to develop peer relationships appropriate to developmental level.
		c	A lack of spontaneous seeking to share enjoyment, interests, or achievements with other people (e.g., by a lack of showing, bringing, or pointing out objects of interest).
		d	Lack of social or emotional reciprocity.
	(2)	Qualitative impairments in communication as manifested by at least one of the following:	
		a	Delay in or total lack of, the development of spoken language (not accompanied by an attempt to compensate through alternative modes of communication such as gesture or mime).
		b	In individuals with adequate speech, marked impairment in the ability to initiate or sustain a conversation with others.
		c	Stereotyped and repetitive use of language or idiosyncratic[2] language.
		d	Lack of varied, spontaneous make-believe play or social imitative play appropriate to developmental level.
	(3)	Restricted repetitive and stereotyped patterns of behavior, interests, and activities, as manifested by at least one of the following:	
		a	Encompassing preoccupation with one or more stereotyped and restricted patterns of interest that is abnormal either in intensity or focus.
		b	Apparently inflexible adherence to specific, nonfunctional routines or rituals.
		c	Stereotyped and repetitive motor manners (e.g., hand or finger flapping or twisting, or complex whole-body movements).
		d	Persistent preoccupation with parts of objects.
B	Delays or abnormal functioning in at least one of the following areas, with onset prior to age 3 years:		
	(1)	Social interaction.	
	(2)	Language as used in social communication.	
	(3)	Symbolic or imaginative play.	
C	The disturbance is not better accounted for by Rett’s Disorder or Childhood Disintegrative Disorder.		

Table 3: Diagnostic criteria, DSM-IV-TR

[2] Unique to an individual.

The diagnosis is completely based on the subjective opinion of the diagnostic physician. There are no acknowledged biomarkers[3] for autism today. Recently, a team of Australian researchers created a genetic test that is able to predict the risk of developing ASD [23]. They identified 237 genetic markers in 146 genes and related cellular pathways that either contribute to or protect an individual from developing ASD.

3.2 Facial emotion recognition

Human communication has two main aspects: verbal such as speaking, singing and sometimes tone of voice and non-verbal such as body language, sign language, paralanguage, touch, eye contact, or the use of text. The atomic units of verbal communication are words and for non-verbal phenomena like facial expressions, body movements, and physiological reactions.

Emotion is expressed via facial movements, speech prosody and text, body and hand gestures, and various biological signals such as heart rate. The number of the facial expressions that we use in our everyday life cannot be strictly specified due to the different surrounding and cultural background that each person has. Ekman's research on the facial expression analysis [24] has focused more on the six basic emotional expressions (fear, anger, disgust, happiness, surprise and sadness) [25]. For these facial expressions, conclusions about their universality have already been made. However, the emotional facial expressions used around the world are much more and some of them are combinations of more than one. There are tons of emotions we express and recognize. Some even overlap. Thus differences in emotion are usually more a matter of degree or intensity instead of kind. In fact, some emotions are merely a blend of others. Plutchik's wheel of emotions shows the basic emotions, their different levels of intensity and their opposites. Some of these relate to Ekman's seven basic emotions, but not all of them have such measurable expression in the faces of humans.

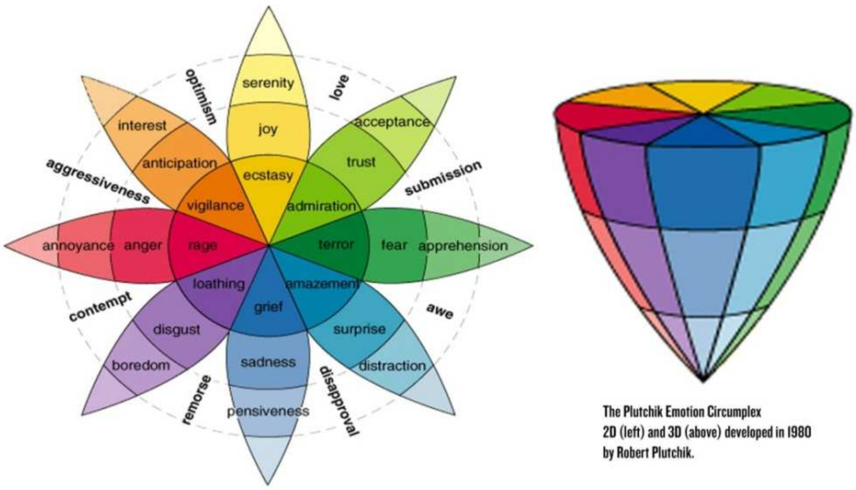


Figure 3: Plutchik's wheel of emotions

Facial expression recognition and classification impacts many different aspects of our daily life. Real life emotions are often complex and involve several simultaneous emotions. They may occur either as the quick succession of different emotions, the superposition of emotions, the masking of one emotion by another one, the suppression of an emotion or the overacting of an emotion. These phenomena are referred as blended emotions. These blends produce "multiple simultaneous facial expressions". Depending on the type of blending, the resulting facial expressions are not identical.

Theories about emotions stretch back at least as far as the Ancient Greek Stoics, as well as Plato and Aristotle. In Aristotelian theory of the emotions, Aristotle with the opportunity to develop his most sustained thoughts on emotions, not only does he define, explicate, compare and contrast various emotions, but also he characterizes emotions themselves.

[3] A specific physical trait used to measure or indicate the effects or progress of a disease, illness, orcondition.

Facial expressions play an important role in our relations. They can reveal the attention, personality, intention and psychological state of a person. They are interactive signals that can regulate our interactions with the environment and other persons in our vicinity. Human emotion is composed of hundreds of expressions and thousands of emotional words, although most of them differ in subtle changes of a few facial features.

The Facial Action Coding System (FACS) [26] is the most commonly used system for facial behavior analysis in psychological research. The FACS is a meta-representation that allows describing facial expression prior to interpretation. FACS detects and measures a large number of facial expressions by virtually observing a small set of facial muscular actions. The system decomposes the face into 46 so called Action Units (AUs), each of which is anatomically related to the contraction of either a specific facial muscle or a set of facial muscles. All facial expressions can be decomposed into their constituent AUs and described by duration, intensity, and asymmetry.

FACS provides a powerful means for detecting and measuring large numbers of facial expressions by detecting a small set of muscular actions which comprise the facial expression. Over 7000 AU combinations have been observed.

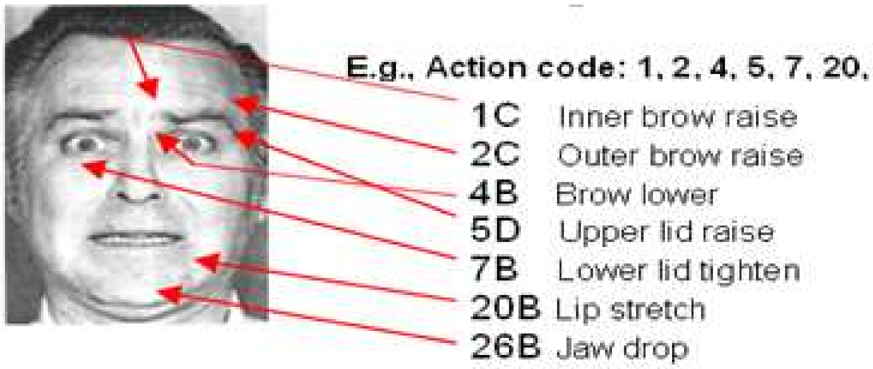


Figure 4: FACS example

A second system, EMFACS, is an abbreviated version of FACS that assesses only those muscle movements believed to be associated with emotional expressions. It is designed to yield limited data about facial behavior using the objective scoring methods of FACS and it is used as a way to reduce scoring time when an investigator is interested only in the emotion signals of the face. As with FACS, in EMFACS what is scored and what is not scored is defined precisely and the coder makes no inference about the meaning of facial behavior. It also requires knowledge of and the ability to identify a) the AUs involved in any facial movement, b) the intensity of the AUs and 3) the degree of asymmetry.

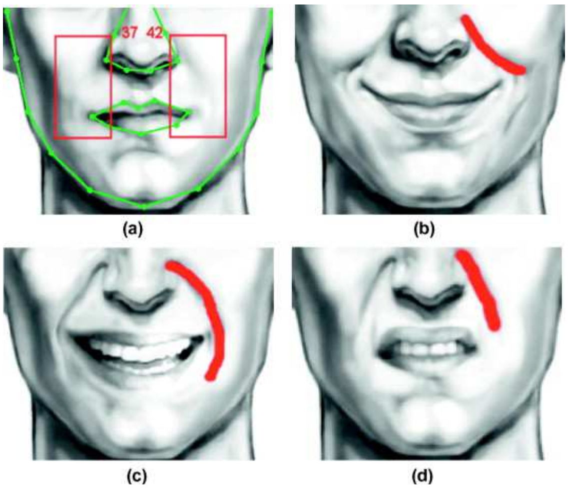


Figure 5: Measure from EMFACS

The recognition of emotion is learned through experience and humans begin to perceive emotions in faces at a young age. Child development, including the ability to recognize emotional expressions, can be fast in the first year of life. It is likely that an interaction of several factors leads to the ability to recognize emotional expressions in faces. The ability to decode facial expressions of emotion is fundamental to human social interaction.

Typically developing children start understanding emotion very early. Five months infants can categorize and recognize facial expressions such as smiling [27]. Evidence also demonstrates face processing skills to be present in infants aged 4 to 9 months [28]. The ability to recognize emotions from facial expressions develops gradually with the age [29]. Processing emotion and decoding facial signals (like emotional expressions) emerge before their second birthday. At the age of 2 years they are able to recognize and use terms for the basic emotions. By this age they also talk about self and other emotional states. From 2 to 5 they expand their emotion vocabulary and they can express more complex emotions [30]. The ability to recognize emotions from facial expressions is improved gradually with the age and a good accuracy level seems to be acquired by 10 years of age [31]. Full proficiency level is reached only after puberty.

Difficulties in understanding and expressing emotions [32] can lead in social interaction impairments, which are a core feature of ASD. Social interaction is the mutual influence between individuals, during which people exchange verbal and nonverbal messages, in order to provide and receive information for themselves and for the others. It is a process of production and exchange of signs that usually affects the behavior of people involved in it. Marked impairments in the use of nonverbal behaviors such as facial expressions, according to DSM-IV-TR, can be responsible for qualitative impaired social interaction. Nonverbal communication is the expression of emotions, moods, attitudes and the general inner world expressed through the body. It constitutes the largest part of a communication process and facilitates clarification between communicators to define the possible role that each of them has at each moment during this transaction. Source of nonverbal signals is the human body and particularly the face, the eyes, the postures, the body movements and the gestures.

3.2.1 Facial features and emotions

Facial expressions are able to express emotions and moods. A facial expression is one or more motions or positions of the muscles in the skin. There are six classically defined facial expressions: Anger, Fear, Disgust, Happiness, Sadness and Surprise. Those basic emotions have been scientifically proven to have a certain facial expression associated with each of them. The Figures 5-7 show the diagnostic features of emotional expressions.

Anger features are: eyebrows pulled down, upper and lower lids pulled up, margins of lips rolled in and lips may be tightened. Fears features are: eyebrows pulled up and together, upper eyelids pulled up and mouth stretched.

Figure 6: Anger (left) and Fear (right) facial features

Disgust features are: eyebrows pulled down, nose wrinkled, upper lip pulled up and lips loose. Happy features are: muscles around the eyes tightened, “crow’s feet” wrinkles around eyes, cheeks raised and lip corners raised diagonal.

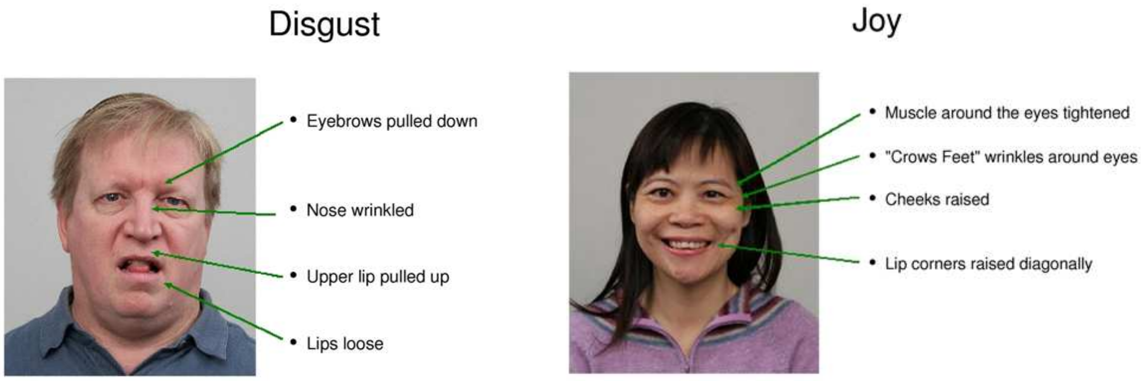


Figure 7: Disgust (left) and Happy (right) facial features

Sadness features are: inners corners of eyebrows raised eyelids loose and lip corners pulled down. Surprise features are: entire eyebrow pulled up, eyelids pulled up and mouth hangs open.

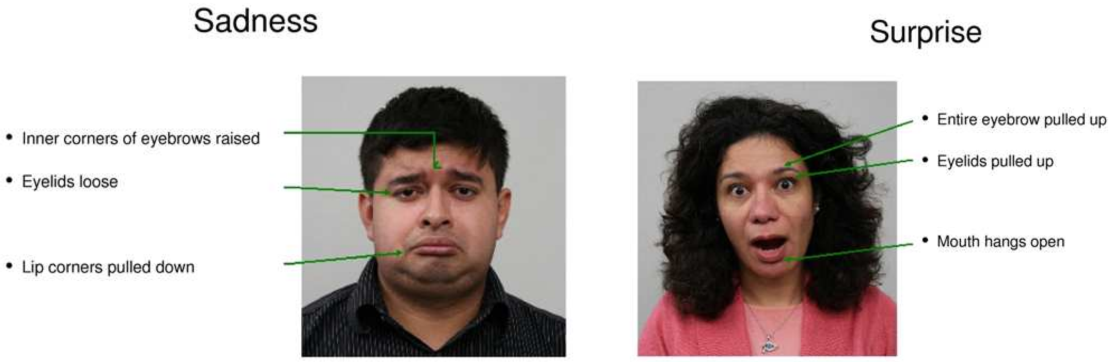


Figure 8: Sad (left) and Surprise (right) facial features

3.2.2 Autism and emotion recognition

Facial expressions give important clues about emotions and provide a key mechanism for understanding, identify and conveying them. Facial expressions underline the feelings of emotions. Autistic children often fail to recognize the qualitative differences and associations between the various expressions of emotions [33]. Due to limited social and emotional understanding they do not know how to adequate interact with other people and this sometimes leads to inappropriate behaviors. Studies have reported that young children with autism experience difficulties recognizing expressions and adults with autism are not as good as typically developing adults at emotion recognition [34]. It has also been recorded that individuals with autism are significantly impaired and exhibit heterogeneity in their emotion processing [35], [36], have difficulties in recognizing emotions from the upper part of the face [37] and exhibit a deficit in overall emotion recognition from facial expressions [38]. Toddlers with autism focus their attention on a single facial feature and treat people as objects rather than as persons to communicate [39]. Children with autism spent a greater proportion of their inspection time viewing non-feature areas of the faces and spent a smaller percentage of time examining core features such as the nose, mouth, and, in particular, the eyes. Similarly, when the percentage of fixations devoted to core features of the face was taken as the index of performance, the results indicated that autistic participants devoted a small percentage of their fixations to primary facial features. In general, the scan paths of the children with autism seemed erratic, undirected, and disorganized, often reflecting the processing of only one or two relatively unimportant features of the face (e.g., an ear, the chin, or region of the hair line).

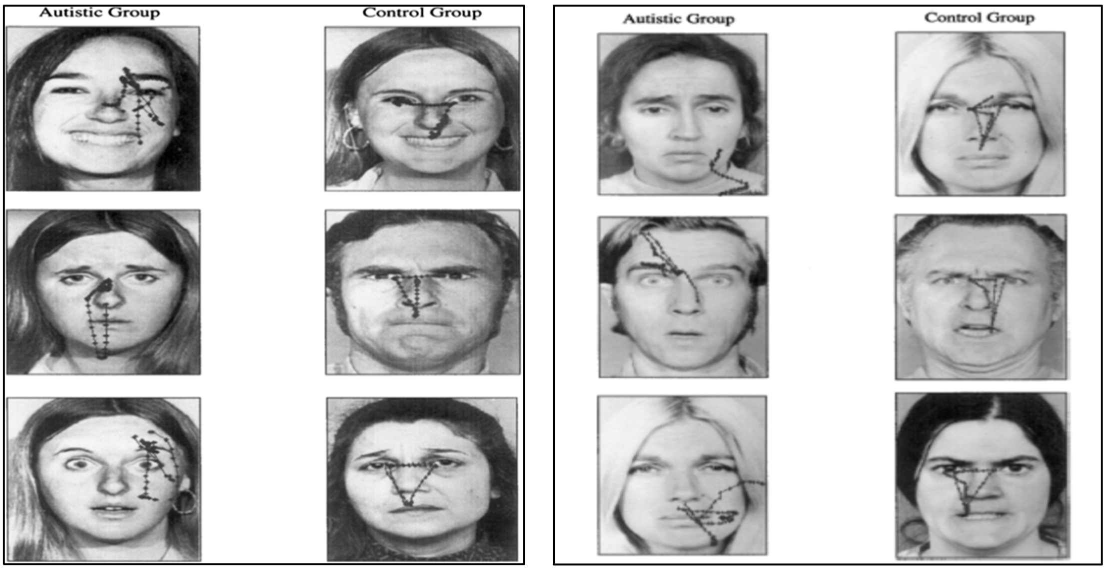


Figure 9: Scan path – children with autism and typically developing

Visual scanning of faces in autism revealed an increased visual fixation time on mouth region and a significant less time on eyes region that was associated with impairment in daily social interaction [40]. In comparison with a typically developing 15-month-old child (matched on nonverbal functioning) and a typically developing 9-month-old infant (matched on verbal functioning), a 15-month-old child with autism showed markedly different patterns of looking. While viewing video scenes of actresses playing the role of a caregiver, the child with autism looked at her eyes less than half as much as either control. The child with autism also spent an increased percentage of time fixating on the actresses' mouths.

Figure 10: Visual scan paths - watching a video

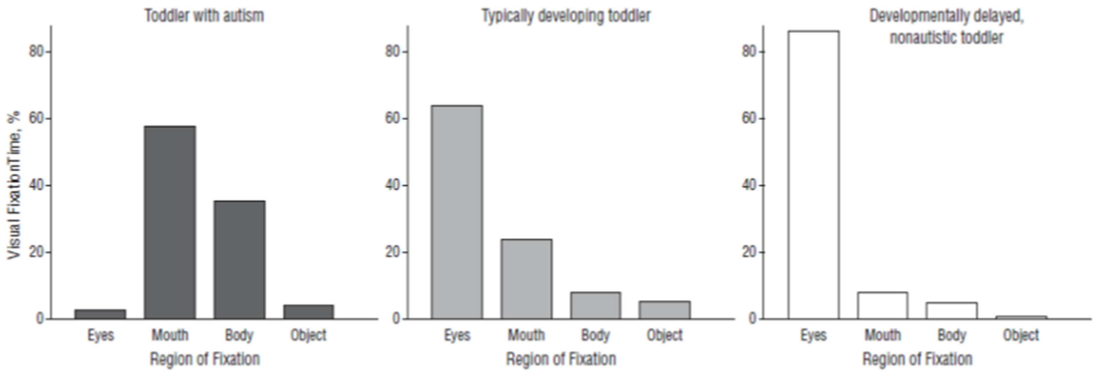


Figure 11: Fixation time summaries - watching a video

There are also evidences that individuals with autism and typically developing individuals decode facial expressions through different mechanisms [41]. A recent study was conducted to evaluate verbal and perceptual skills implicated in the recognition of facial emotions. Individuals with ASD, 5-20 years old, were tested for their perception and identification of facial emotions. Results indicated unimpaired ability to label basic facial emotions and impaired ability to generalize them [42]. Other studies claim that individuals with ASDs do not show impaired emotion recognition [43] and these results indicate the wide variation in the manner that individuals are affected. Autistic children do not suffer a global impairment of facial emotion recognition.

3.3 Treatment approaches

Treatment approaches aim to improve social interaction, conquest communication and control inappropriate behavior. Children with autism are more likely to initiate positive interaction after treatment [44]. Educational interventions and medical management of the symptoms are the two major categories for treatment. Autism requires a particular educational process to assist children with ASD in learning basic skills. Education is also considered as a solution for the socio-emotional deficits and training is claimed to improve face processing abilities and strategies in autism [45]. A variety of educational interventions have been proposed for children with autism and many proponents have claimed developmental improvement and other benefits [46].

3.3.1 Traditional educational interventions teaching emotions

Traditional therapy techniques and tools mostly used for teaching children with autism about emotions are social stories, Developmental Individualized and Relationship based (DIR) / Floortime, Social Communication Emotional Regulation, Transactional Support (SCRET), Applied Behavioral Analysis (ABA), Early Intensive Behavioral Intervention (EIBI), Picture Exchange Communication System (PECS) and Treatment and Education of Autistic and Related Communication Handicapped CHildren (TEACCH).

Social stories developed by Carol Gray [47] are an approach for teaching individuals with ASD social behaviors. They are short stories that describe and give information about social situations or interaction which children may find difficult or confusing. The stories also outline how to respond in such situations. There are guidelines and criteria for writing a social story. Their main goal is to provide accurate social information for improving understanding of events that may lead to more effective responses and not for effecting changes. Social stories about feelings and appropriate emotional responses should provide information about feelings of other characters in the story and the consequences of ignoring those feelings. This can help children with autism to learn how to understand emotions in context and how to cope with them.

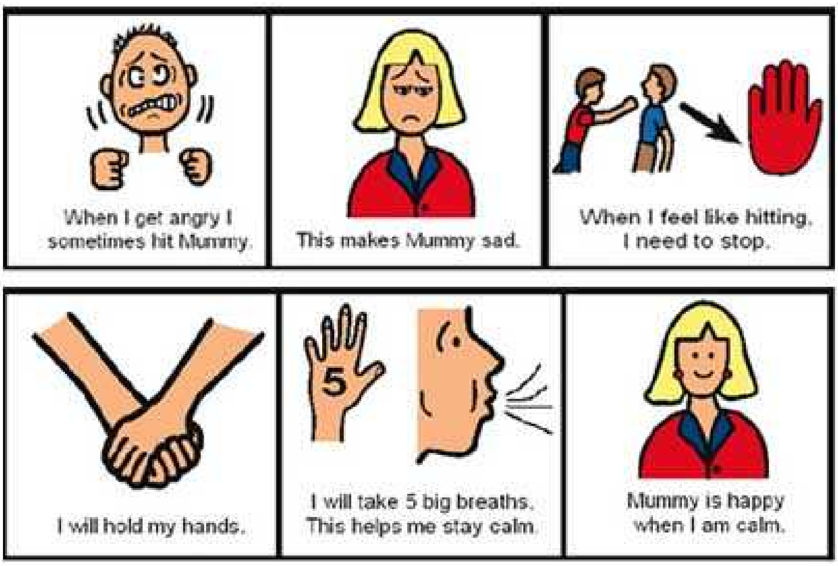


Figure 12: Social story - angry

DIR / Floortime parent intervention [48] can help children with autism to connect emotionally and to build social and intellectual skills. This approach follows the child's emotional interests and provides one to one, intensive, play - based intervention.

SCRET is a framework that incorporates practices from other approaches and evidence-based practices such as the methods previous mentioned [49]. It is a multidisciplinary model to support children with autism to enhance communication and socio-emotional abilities.

The ABA therapy is often used as a treatment for children with autism. It is a widespread and recognized method for increasing learning, communication and appropriate social behavior [50]. Positive reinforcement, reward for desired behaviors and ignorance for inappropriate behaviors are the main characteristics. By rewarding and encouraging desired behavior, children are more likely to repeat it. According to the ABA intervention method, teaching children with autism about emotions can be achieved by providing examples of appropriate emotional behavior and then rewarding when a child gives the correct emotional response.

Young children with ASDs usually receive a home-based program, the EIBI, which is based on the principles and technologies of ABA [51]. Early intervention appears to lessen the effects of autism and children with ASD appear most able to benefit when intervention begins very early (2-4 years old) [52], [53]. Toddlers, who attended an early intervention program until the age of 3, appear to benefit from this, since 31% of those children were functioning in the typical developing range after the intervention [54].

There are presumptions that individuals with autism think in pictures [55]. Children with ASD are reported to have strong visual processing capability and a good performance on difficult visual search tasks [56], [57]. Augmentative and Alternative Communication (AAC), such as PECS, is frequently used to increase functional communication in children with ASD. PECS is an effective visually-based system for communication via icons [58]. Visual supports are also suggested for social skill development enhancement in young children with ASD [59]. Picture cards with cartoon faces or illustrations of people faces or photos showing different facial emotional expressions reinforce and support learning. Images showing only the face against a plain background and clearly depicting a single emotion, usually used to help children with autism to understand and recognize basic facial expressions. Those cards may be used by calling out the name of an emotion and challenging children to give you the correct card or asking them to look at the face and reflect back the emotion displayed. Cards show people in real situations with natural settings and contexts provide visual stimuli for discussion about a range of emotions or about what people feel, why they may feel that way and possible responses to these feelings.

Figure 13: PECS emotion cards

Also programs that emphasize visual support strategies are commonly used in education classrooms. A well-known and widely applied model, the TEACCH, is a teaching strategy which emphasizes in structured and predictable learning environment. The program utilizes visual cues to increase independence and to teach new skills, such as facial emotions, to children with autism. It involves daily schedules, visual materials, individualized treatment and parental support [60]. The main goal of TEACCH is to help children with autism grow up to their maximum ability by adult age. It is a structured technique specialized to the person's visual processing strengths by organizing the physical structure of the room and providing a visual conduct to supply information about activities. Structured teaching places heavy reliance upon teaching through visual means due to the difficulties that children with autism have with processing verbal information. Visual structure is provided at a variety of levels such as organizing areas of the classroom, providing a daily schedule using pictures or written words, as well as visual instructions and visual organization signaling the beginning and end of tasks. The technique is based upon the observation that children with autism learn and connect information differently than other children. It assumes that many inappropriate behaviors of children with autism are the result of difficulty understanding what is expected of them.

3.3.2 Computer interventions teaching emotions

Many children with autism find computers very engaging and computer games strongly attract them in a positive way. Computers are rich, stable, predictable, consistent teaching tools and they provide a proper and more motivational learning environment for them [61]. They are great educational tools since children with autism often experience discomfort with unpredictable social environment and they prefer a controlled learning environment [62]. Autistic individuals can enjoy learning and improve their skills with computer-based intervention. Nowadays computers are the most adaptable assistive technology devices available for children with autism and various computer games have been developed to help them to manipulate their impairments. Some of those games focus on their social interaction training and especially learning about emotions [63]. Although effects on social and emotional skills are mixed, computer intervention is a promising practice.

The Emotion Trainer [64] is a multimedia computer program for improving school age students' ability to recognize and predict emotions in others. It has positive effects on users' understanding of emotion, particularly with repeated use.

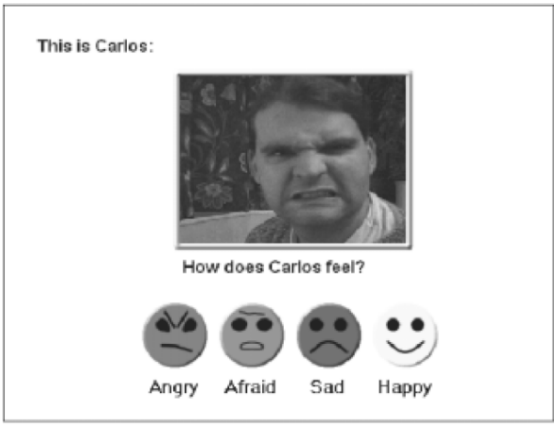


Figure 14: Emotion Trainer computer program

Mind Reading [65] is an interactive systematic emotion guide for teaching individuals with Asperger syndrome. Experiments show that Mind Reading is effective in teaching adults with Asperger syndrome or high-functioning autism to recognize complex emotions.

Figure 15: Mind Reading game

Another computer training program for teenagers with autism is the "What to choose" game [66]. It is software with human and cartoon facial expressions, 3D images, text and audio for training individuals with autism in understanding dialogues that contain pragmatic subtleties. Children with autism have failed to take into account the causal link between facial expressions and the outcome of the dialogue.

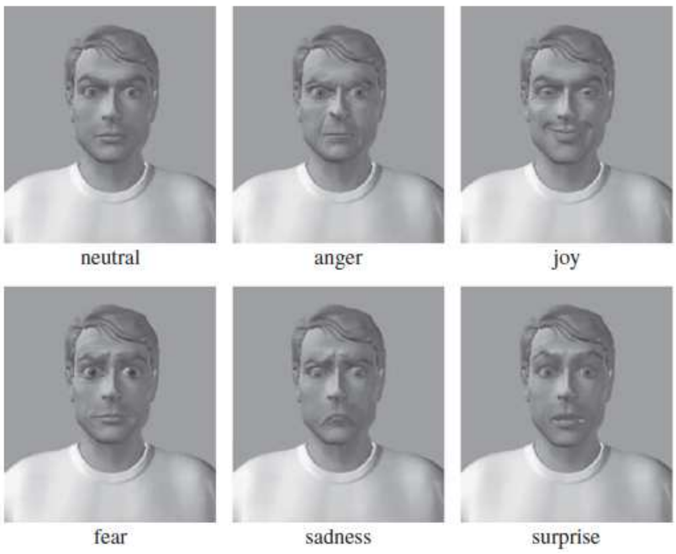


Figure 16: What to choose game

cMotion [67] is a computer game that uses virtual characters to teach emotion recognition and programming concepts to children with autism. The game is designed to teach the users how to recognize facial expressions and manipulate an interactive virtual character by using a visual drag-and-drop programming interface.

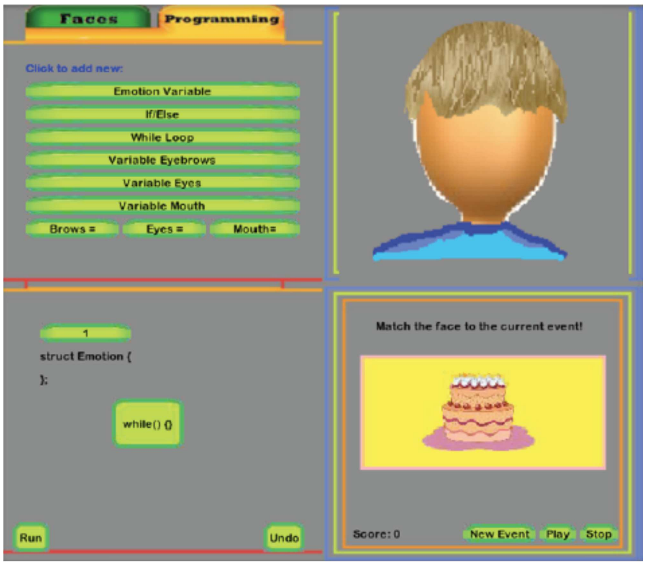


Figure 17: cMotion game

A computer-based intervention for face training is the Let's Face It! (LFI!) program [68]. This program is comprised of seven interactive computer games that target the specific face impairments associated with autism. Those games are organized into a theoretical hierarchy of face processing domains. The Let's Face It! Program shows promise as an effective intervention tool and treatment alternative.

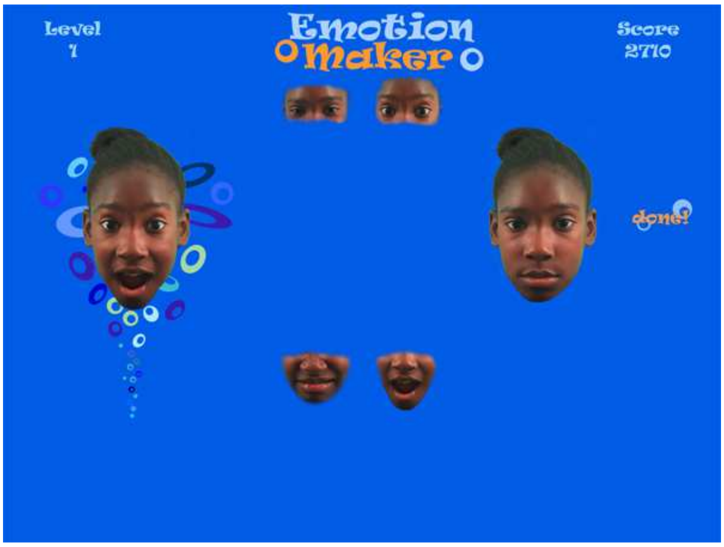


Figure 18: Let's Face It! game

Emotion Mirror is a project that integrates Computer Expression Recognition Toolbox (CERT) with the program Let's Face it! [69]. It is a computer assisted intervention system to enhance facial expression perception and production in children with ASD. This game could be beneficial for children who are already have learned labeling and understanding of emotions.

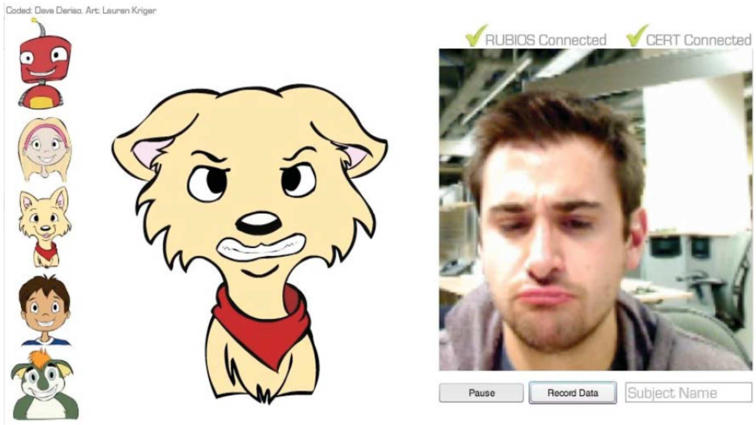


Figure 19: Emotion Mirror game

The LIFEisGAME is a facial emotion recognition learning system based on the interaction between humans and 3D avatars [70]. It is a computer-based approach that uses real time facial synthesis of 3D characters in order to teach autistic people to recognize emotions from facial expressions.

Figure 20: LIFEisGAME game

3.3.3 Other interventions teaching emotions

MIT Media Lab proposes a wearable device [71] that perceives and reports on social-emotional information in real-time human interaction. The system records and analyzes the facial expressions and the head movements of the person with whom the wearer is interacting. This wearable platform is suggested as an exploratory and monitoring tool to assist individuals with ASD in perceiving communication in a natural environment.

The Transporters [72], a DVD for teaching emotion recognition, is an animation series for children with ASD (preschoolers or with significant learning difficulties). It involves toy vehicles and real life faces of actors that show emotional expressions in social context. DVD intends to explore whether creating an autism-friendly context of predictable mechanical motion can introduce facial expressions of emotion that then can be learned more easily than is possible in the real world. The aim of this program was to teach not just some basic emotions but also some more complex ones. Evaluations of this project showed improvements in emotion comprehension and recognition skills [73], [74] but limited efficiency in teaching basic emotion recognition skills to young children with autism with lower range of cognitive ability [75].

Figure 21: The Transporters DVD

PixTalk [76] is a software application for Windows Mobile Smart-phones which can be used as part of ongoing therapy. Teachers and caregivers can access a web site and select from an online library the images to be downloaded on to the Smart-phone. Children also can browse and select images to express their intentions, desires, and emotions using PixTalk. An accompanying website designed to help children with autism spectrum disorder to communicate. It is based on PECS in which children use visual symbols as a form of communication. Children can browse and select images using the touch screen application to express their intentions, desires and emotions. PixTalk can be downloaded for free and be used on any Microsoft Windows smart-phone or PDA.

Figure 22: PixTalk game

4. Final problem statement

Many individuals with autism show deficits in their perception and recognition of facial emotions. With the appropriate intervention, elimination of those deficits can be achieved. Interventions are proposed to start as early as possible and children with ASD appear most able to benefit when intervention begins at 2-4 years old. Typically developing children by the age of 2 use term for basic emotion and from 2-5 years old they expand their vocabulary. So, for children with ASD we suggest the age of 2-6 (preschool) to be set as the starting level of the emotion teaching process.

Difficulties in identifying and describing feelings are assumed to be an integral part of autism. Educational interventions can be used as a tool to help individuals with autism to cope with those deficits. As the number of children diagnosed with autism increased, new methods for educating this population become necessary. In several studies, computer-based programs have been widely used with success to teach people with ASD to recognize emotions. However, those computer interventions require considerable skills for interaction as the users have to control a mouse or a keyboard. Such abilities are beyond very young children with autism as they have major restriction in their efficiency to interact with computers. Our aim is to design and develop a gesture controlled Serious Game, as an early intervention, for Greek preschoolers (2-6 years old) with autism to teach them basic facial emotions. Our approach takes into account the specific characteristics of preschoolers with autism and their physical abilities. By creating an educational computer game which proposes physical interaction, we hope to make early intervention more appealing and to foster learning of emotions. This software intends to support individuals with autism, their families and their trainers and also to assist the preschool teaching and learning process. In this context, we conclude to the following Final Problem Statement:

"How can innovative technologies help preschoolers with autism to improve facial emotion recognition skills?"

5. Analysis

Nowadays, there are interfaces created to be natural. Those interfaces can be useful and more appealing for young children and individuals with autism. They provide interaction with the system in a physical way. Users can control the system with gestures or their body movements. They don't have to wear anything and they also don't have to touch any physical input device. A natural interaction device which operates in the context of NUI can provide interaction in a physical way.

Previews studies that used Serious Games with children with autism have taken for granted the required skills and in particular the ability to use the mouse or the keyboard. People with ASD demonstrate delay in fine motor skills which causes difficulties in grasping and manipulating objects, such as mouse [77]. Many individuals with autism experience motor control problems in their hands, leading to difficulties in operating the computer mouse. The difficulties they encounter include moving the computer mouse to designated area, corresponding to the location of the cursor to the mouse movement, pressing down and lifting up fingers for the clicking motion, and clicking the correct button on the mouse to make selection [78]. Those difficulties are also found in typically developing individuals. Very young children (autistic and non-autistic) are unable to functionally operate the computer mouse. Experiments showed that four and five year old non-autistic children make more and less accurate mouse sub-movements on approach to targets than young adults [79]. With increasing age children in this study demonstrated significantly faster and more accurate use of the mouse think that provides evidence of a strong link between speed and accuracy skills and developmental maturity. The same problems occur when players are required to use other game controllers such as joysticks and gamepads. Various research studies have been conducted to evaluate game controllers with the use of ISO 9241-9 [80] based on Fitts' law tasks which is an effective and widely used predictor of performance and comfort. For point and select task, evaluation showed that participants prefer using a mouse and results indicated that remote pointing devices perform poorly in terms of throughput, speed and error rate when compared to a mouse [81]. Touchpads are consider to be a good solution for young children but evaluations have also shown inferior performance, increased movement time, lower throughput and higher error rates when compared with a joystick [82], a trackball [83] and a mouse [84]. Therefore, we propose a physical interaction based on gestures and a controller-free interface as a different interaction technique. Our game will allow the user to employ gestures for navigation and interaction.

5.1 Serious Games

Also currently, ample researches in Serious Games for children with autism have been done. Serious Games are games designed for a primary purpose other than pure entertainment, fun or enjoyment [85]. Michael Zyda defines a Serious Game as: "a mental contest, played with a computer in accordance with specific rules that uses entertainment to further government or corporate training, education, health, public policy, and strategic communication objectives." [86].

Serious Games for autism cover matters related to education, therapy for communication, psychomotor treatment and social behavior enhancement [87]. Autism Serious Games for education are designed to help teachers or student during the teaching and/or learning process. Studies show that Serious Games are very effective in the areas of therapy and education for children with autism.

Many approaches to technology-enhanced intervention rely on educational methods shown to result in good outcomes. Modern computer games show potential not just for engaging and entertaining users, but also in promoting learning. Game designers employ a range of techniques to promote long-term user engagement and motivation. These techniques are increasingly being employed in so-called serious games, games that have non-entertainment purposes such as education or training.

Computer interventions such as Serious Games could be a good approach as they also have positive effects on children with autism. Serious Games have the potential to support the process of learning and to motivate individuals to learn more. They usually run on a personal computer and they provide a different style of learning. Although serious games are traditionally associated with software developments, developing them in the autism field involves studying the associated technology and paying attention to aspects related to interaction with the game. We will attempt to apply a Serious Game approach to teach children with autism facial emotion recognition.

5.2 Gestures

Gestures play an important role in human-to-human communication, and they can also be used in interaction with computer systems. As computers are responsible for an increasing number of tasks in the society, human computer interaction (HCI) is becoming more important. Typically HCI is done with keyboard and mouse, which provide a stable and familiar way to access computers. Nevertheless, in some cases they cannot be accessed or they do not provide the best possible way of HCI because the interaction with them is too slow. Modern interaction devices, such as Microsoft Kinect, Nintendo Wii (Wii) and multi touch surfaces provide an interesting and more natural way for HCI. The interaction devices by themselves open a channel to access the computer, but the challenge is how to use them in the best possible way.

One goal of HCI research is to increase the naturalness of interaction. By increasing naturalness, researchers think of using similar methods in interaction that humans use in communicating with each other. The most natural way of human communication is the verbal communication. It is being increasingly used as a method for interaction. For example Microsoft Kinect provides speech recognition equipment, which consists of equipment for capturing audio and tools for processing voice. The second main human-to-human communication method is the nonverbal communication. Non-verbal communication concludes all of the other communication methods that humans use, such as facial expressions or hand gestures. In HCI the most common way of mimicking human-to human nonverbal communication is the gesture-based interaction.

Gestures can be divided into two groups of static and dynamic gestures. A static gesture is a single pose formed by a single image, while a dynamic gesture consists of movement and forms from a sequence of images.

5.2.1 Gesture recognition

The technique of capturing gestures by computer is called gesture recognition. Gesture recognition uses mathematical interpretations of human motions by a computing device and is used to trigger actions on the computer. Technology enabling gesture recognition consists of two main types of devices: contact-based devices and vision-based devices. Contact-based devices include: multi-touch screens, accelerators, controllers and instrumented gloves. Vision-based technology is based on one or several cameras. The captured video sequence is observed in order to analyze and interpret the motion of it. Various vision-based sensors include: monocular cameras, body markers and infrared cameras.

The methods for gesture recognition vary a lot. It can be done through feature extraction and statistical classification methods. These methods consist of two stages. In the learning stage, the extracted features are categorized. In the classification stage, the movement is compared to learned features. In model-based methods the recognition process happens in a single stage where the target's parameters are extracted and then filled to the adequate gesture model. In template matching methods, the whole gesture is considered a template, instead of using either feature extraction or a gesture model. Hybrid methods are combinations of these methods.

5.2.2 Gesture-based interaction

HCI done with gestures is called gesture-based interaction. Gestures allow direct, natural, and intuitive way of interaction. One benefit of using gesture-based interaction is that it makes a wider range of actions available to manipulate the system compared to traditional interfaces. Other benefit is its interface's ability to change any time, allowing it to be more customizable for application's needs than traditional interfaces.

Gesture-based interaction is a more advanced and a more natural way of interaction than the interaction with a traditional keyboard and mouse. Most gesture based interaction approaches use intensity images to extract necessary motion information. However, recently 3D motion information has been introduced for recognizing 3D gestures. To be able to recognize gestures in three dimensions, the system requires some form of vision-based device or a controller, which can track the position of the user. Three dimensional gestures can be performed with the full body. For the recognition of human movement, the computer needs a model or an abstraction of the motion of the human body parts. Three dimensional gestures are applied widely in the game industry, most notably with Microsoft Xbox Kinect.

5.3 Natural User Interface

NUI is a user interface that is easy to use and as seamless as possible. The perception of using these types of interfaces can be faded out by not appending any interaction devices to the user and by designing the communication process in a form that is really natural and understandable for the user.

The natural user interfaces are an emerging paradigm in human-computer interaction which concentrates on human skills such as touch, vision, voice, motion and higher cognitive functions such as expression, perception and recall. It aims to take advantage of the power of a larger breadth of communication modalities, leveraging the skills that people earn through traditional physical interactions.

5.4 Microsoft Kinect

Kinect (formerly called "Project Natal"[4]) is a motion sensing USB (Universal Serial Bus) input device by Microsoft that enables user to control and naturally interact with games and other programs without the need to physically touch a game controller or object of any kind. Kinect achieves this through a NUI by tracking the user's body movements and by using gestures and spoken commands.

Launched in November 2010, Kinect uses technology by Israeli company PrimeSense that generates real-time depth, color and audio data of the living room scene. Kinect works in all room lighting conditions, whether in complete darkness or in a fully lit room, and does not require the user to wear or hold anything.

Figure 23: Kinect sensor

Kinect is a horizontal bar connected to a small base with a motorized pivot and is designed to be positioned lengthwise above or below the computer or TV screen. The device features an RGB (Red Green Blue) color camera, a depth sensor, and a noise-cancelling multi-array microphone.

Kinect is capable of simultaneously tracking two active users. For full-body, head to feet tracking, the recommended user distance from the sensor is approximately 1.8 m. Because Kinect's motorized tilt mechanism requires more power than what can be supplied via USB ports, the device makes use of a proprietary connector and ships with a special power supply cable that splits the connection into separate USB and power connections, with power being supplied from the mains by way of an AC/DC adapter.

[4] http://www.microsoft.com/en-us/news/features/2010/jun10/06-13kinectintroduced.aspx

Figure 24: Kinect AC/DC adapter

The Kinect for Windows sensor is available in the United States, for a suggested retail price of US$249. In addition, the Kinect for Windows sensor is available through Microsoft distributors in Australia, Austria, Belgium, Brazil, Canada, Chile, China, Colombia, Denmark, Finland, France, Germany, Greece, Hong Kong SAR, Hungary, India, Ireland, Italy, Japan, Korea, Mexico, the Netherlands, New Zealand, Norway, Poland, Portugal, Russia, Saudi Arabia, Singapore, South Africa, Spain, Sweden, Switzerland, Taiwan, the United Arab Emirates, and the United Kingdom. In Greece users can buy Kinect for Windows for 249.99 €[5]. This sensor system is relatively cheap compare with other depth camera which makes it highly accessible for research purpose.

Using the Microsoft Kinect for assessment of 3D anatomical landmark positions provides both benefits and drawbacks when compared to a marker-based 3D camera system [88]. What really differentiates Kinect from other devices is its ability to capture depth. The device is composed of multiple sensors. In the middle it has a RGB camera allowing a resolution up to 1280x960 at 12 images per second. The usual used resolution is 640x480 pixels at 30 images per second maximum for colored video stream as the depth camera has a maximum resolution of 640x480 at 30 frames per second [89]. On the far left of the device, it has the IR light (projector). It projects multiple dots which allow the final camera on the right side, the CMOS depth camera, to compute a 3D environment [90].

Figure 25: IR light dots

[5] http://www.multirama.gr/products/product?pid=0721874&catid=070305

For the audio inputs, the device has 2 microphones on each side for voice recognition. The device is mounted with a motorized tilt to adjust the vertical angle. Kinect can detect up to 2 users at the same time and compute their skeletons in 3D with 20 joints representing body junctions like the feet, knees, hips, shoulders, elbows, wrists, head, etc.

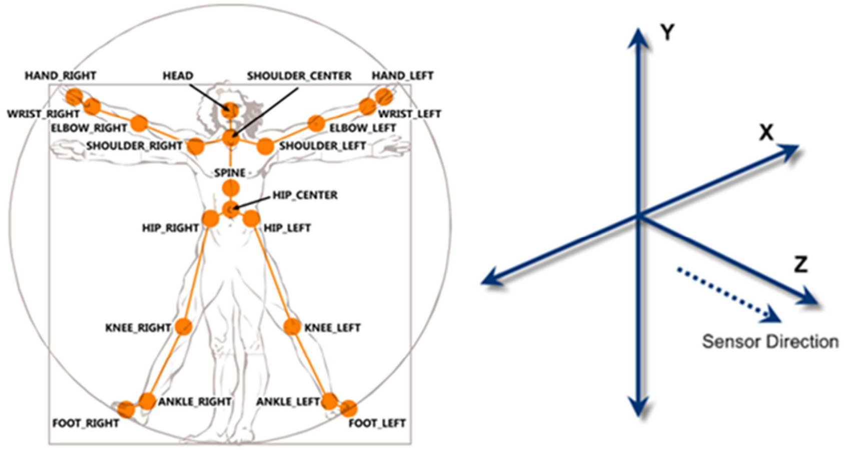


Figure 26: Joints tracked by the Kinect in the 3Dreferential.

Kinect tracking area is in front of the sensor. The field of view is determined by depth camera settings. In default mode the Kinect is capable of seeing people standing between 0.8 m and 4 m, but as the users need to be able to use their arms, the practical distance of default mode is 1.2 m to 3.5 m. The horizontal and vertical field of view in default mode is illustrated in Figure 26.In near mode the Kinect is able to detect standing people from 0.4 m to 3 m, with a corresponding practical range of 0.8 m to 2.5m.

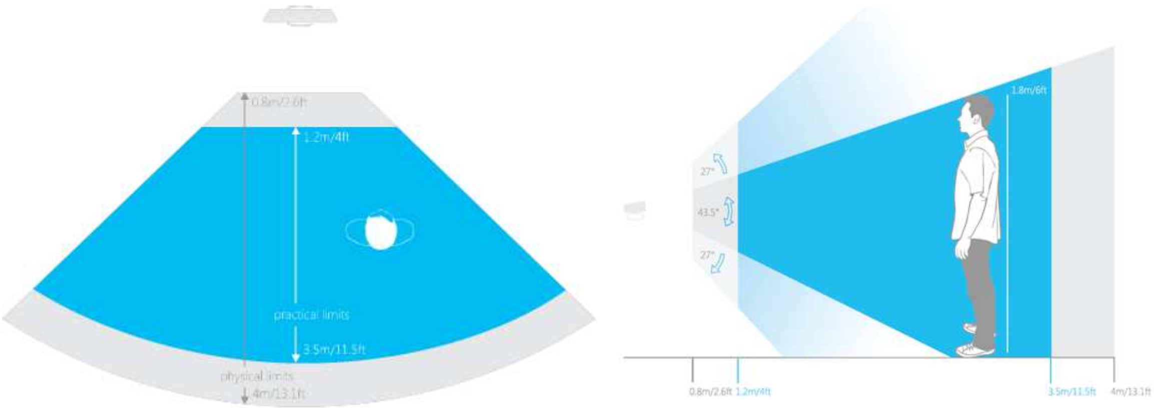


Figure 27: Kinect horizontal and vertical field of view in default range

Kinect's impact has extended far beyond the gaming industry. With its wide availability and low cost, many researchers and practitioners in computer science, electronic engineering, and robotics are leveraging the sensing technology to develop creative new ways to interact with machines and to perform other tasks, from helping children with autism to assisting doctors in operating rooms.

5.4.1 Kinect-based applications

The first applications that have been developed are related to controller-free video games that are based on user's silhouette extraction and skeleton tracking. The extrapolation and tracking of user's movements has been successfully used in efficient physical rehabilitation platforms. Kinect have been used to develop home-based exercise programs for patients with chronic pain [91]. A such rehabilitation game targets people with chronic low back or neck pain, providing an adventure game through which users progress by treadmill walking, reaching, relaxing the trapezius muscle, and moving/rotating the cervical spine.

Kinect-based gesture recognition systems have been integrated in several applications such as: sign-translation platform for hearing-impaired people [92], contact-free visual data interaction in operating rooms [93] and human-robot interaction [94]. Others active computer vision research areas where the Kinect features are very attractive are robot-navigation systems [95] and 3D environment reconstruction [96].

Figure 28: Controller-free interactive exploration of medical images through the Kinect.

The main reasons for using a Kinect camera for above mentioned researches is because Kinect is a low cost and powerful device which can extract 3D information accurately while still providing that other RGB camera can achieve.

6. Design

In this project we present an educational computer-based single player game specially designed for preschoolers with autism. The primary objective of this game is to assist young children with ASD to identify different emotions from facial expressions. The main goal of this game is to provide recognition and understanding of facial emotions through early intervention.

Educating children with ASD and particularly teaching them to recognize and identify emotions from facial expressions is a significant challenge and a complex task. The use of educational software for teaching social and emotional skills could help students with ASD to improve those abilities [97]. Learning tasks developed in digital environments using information technology can motivate the desire to learn in ASD students. As we have already mentioned in the Pre-analysis, computer interventions appear to be particularly appropriate for people with ASD for several reasons. Traditional interventions previously mentioned can be transferred into a computerized version. In our game, we will provide one to one intensive play-based intervention, visual support, positive reinforcement and reward, structural and predictable learning environment and we will incorporate practices from all traditional approaches already mentioned in order to make intervention more efficient.

We based the development of the game on some main principles. As a Serious Game, it should have an impact on the player in a real life context [98]. Our game is explicitly designed to reach a specific purpose beyond the game itself. It aims to teach preschoolers with autism, facial emotion recognition so as to enhance their social interaction.

Taking into account specific characteristics of autism, we designed the game in order to meet the needs of students with ASD. A recent study conducted to analyze user needs for a Serious Games for teaching children with ASD emotions, revealed the characteristics of the children's game play behaviors [99]. The observation showed repetition, matching instead of learning the features, lack of holistic face processing and deliberately incorrect selection. Count on those findings, our game was intentionally designed to avoid those behaviors. The emotions will be presented automatically, one by one and in different order. For repetition, we made the choice not to give them the opportunity to choose the same emotion again and again. For matching instead of learning the features, we decide to describe the features of facial expressions in a separate level. For lack of holistic face processing, we made the choice to describe all the face features that reveal the emotion and to ask them to look at each feature separately. For the deliberate incorrect selection, we decide nothing special happening when they give a wrong answer.

Our design also incorporates a theory-driven game design framework supported by learning and developmental theories [100]. The framework is based on the integration of Kolb's experiential learning model and Piaget's cognitive model. From this systematic approach were extracted six essential elements for designing games to teach children with ASD emotions. Those elements are: matching, recognition, observation, understanding, generalizing and mimicking. We took those elements into consideration during the design process.

6.1 Game environment

The game environment is simple and less detailed in order to avoid children's distraction. Individuals with autism are reported to have sensory abnormalities and to have enhanced perception of details [101] which may causes distraction. Individuals with ASD have significantly better visual acuity compared with control subject's acuity, so superior that it lies in the region reported for birds of prey. For those reasons we select black context presented on a white background and grayscale stimuli. Black and white contrast may also help to increase and retain child's attention and keep them focused on the screen.

Game begins with an instruction page where the child is informed what is going to happen, what he/she has to do and how he/she can do it. Apart from the text on the screen, audio instructions are also provided. Audio cues are important as the information presented is clear and age-appropriate. When the child feels ready, he/she can choose to start the game.

The game provides a structure learning environment which consists of 3 different levels with increasing difficulty. Breaking the teaching intervention into small learning steps makes the task easier to perform. In the first level children should learn labeling emotions by correlating emotion terms with images. In the second level they should learn to recognize emotions from their description and their association with facial features. In the third level they should learn to identify the causes of various feelings in different situations, obtained through the use of social stories. Those three levels provide recognition, matching, observation, understanding and generalization of facial emotions.

Individuals with autism are usually visual learners, which mean that they understand written words, photos and visual information better than spoken language. Information is good to be presented through their strongest processing area. When teaching individuals with autism about emotions, it is important to keep explanations as simple and as concrete as possible. It is also recommended to describe each feeling pictorially by using pictures with clear outline, minimal details and color [102]. For young children it is advisable to keep to the basic emotions. In our approach, the basic emotions selected include happy, sad, angry, scared and surprised. Those emotions were chosen because typically developing children can recognize and understand them between 2 and 7 years of age. The face stimuli we used are 15 grayscale photographs of male and female faces, taken from the CAlifornia Facial Expressions (CAFE) dataset [103]. All images used in the game, met FACS criteria [26] and all of this faces have been certified as "FACS-correct" [104]. The stimuli are presented on each trial with different pair of the same size (300x250) photos and the goal is to choose the correct image among the two. We allow player to examine the stimuli for as long as necessary before provide a response.

A B C

Figure 29: CAlifornia Facial Expressions (A) angry – (B) happy – (C) sad

All the stimuli were converted from .pgm files to .png so they can be opened by the program. We also used Matlab to change stimulus intensity in order to make them more effective. We darken the images by subtracting a constant value from each pixel value.

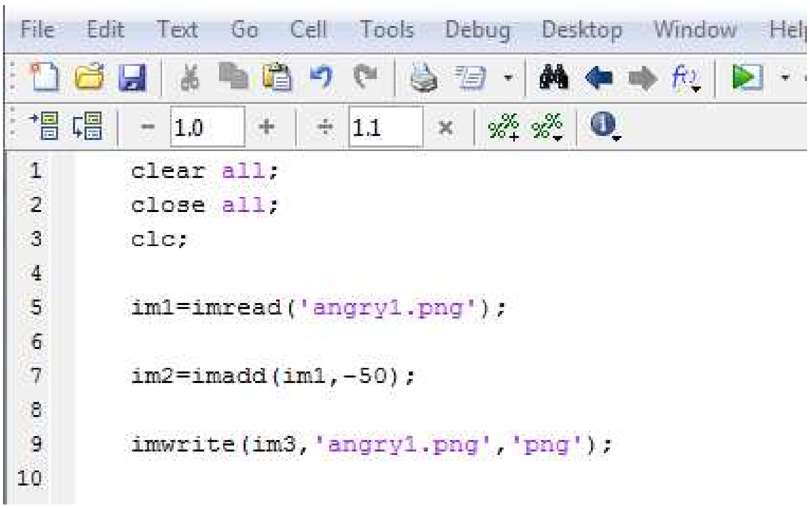


Figure 30: Image processing with Matlab

6.2 Story board

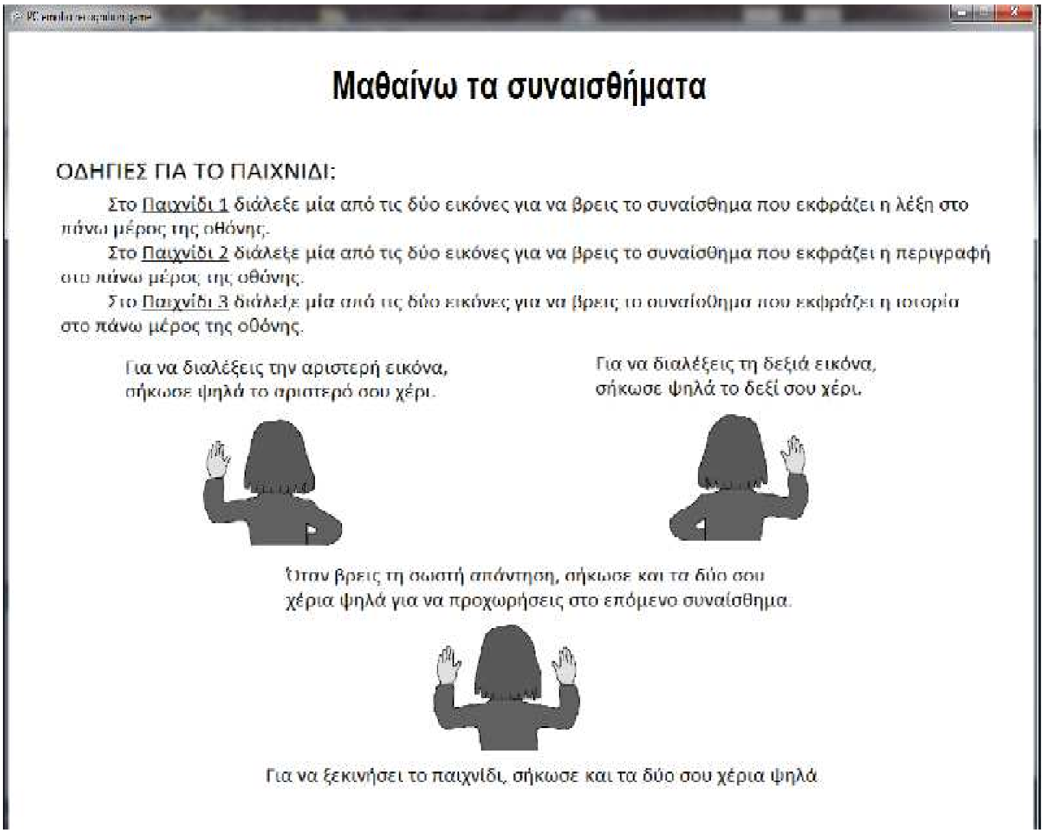
The game consists of five screens where players can navigate during the game play. From the start screen, player can only move to Game 1, from Game 1 can only move to Game 2 and so on. The same approach has been adopted for the whole game. Game 1 consists of five parts where each one of them has a different emotion. Players can move on only forward from first emotion to the second and so on.

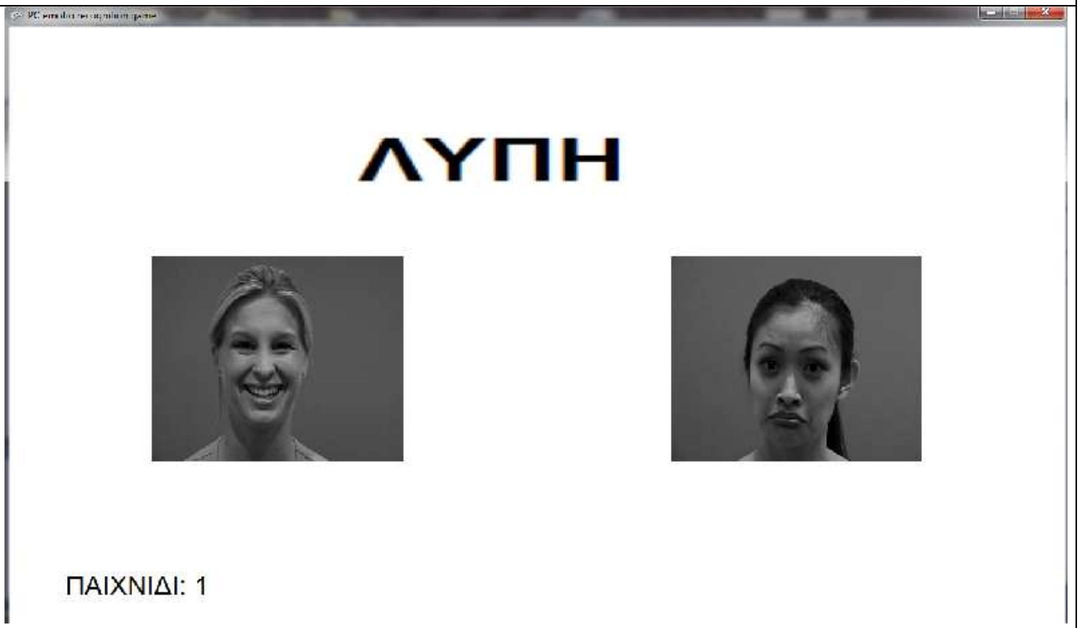
Figure 31: Game Process

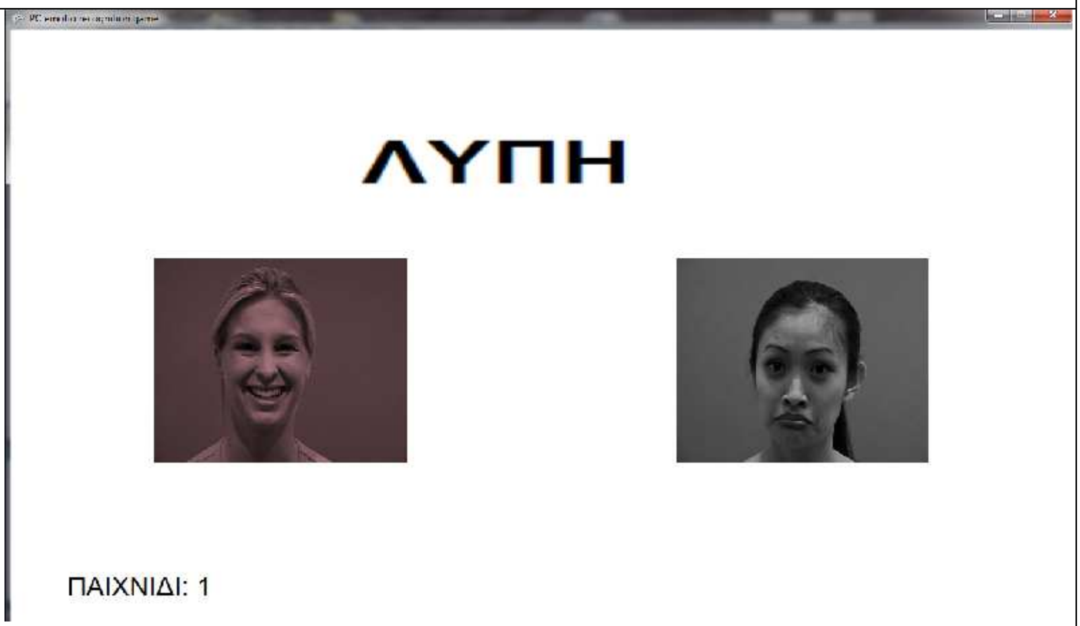
Our storyboard provides a full description of the game. Storyboards are another way to explore narrative possibilities or to rehearse a performance. A storyboard is a series of connected pictures, with or without words, that tells a continuous story. Storyboarding is used in software development as part of identifying the specifications for particular software. During the specification phase, screens that the software will display are drawn, either on paper or using other specialized software, to illustrate the important steps of the user experience. Storyboarding a game means the maker sketches out ideas for characters and scenes and then places them in order.

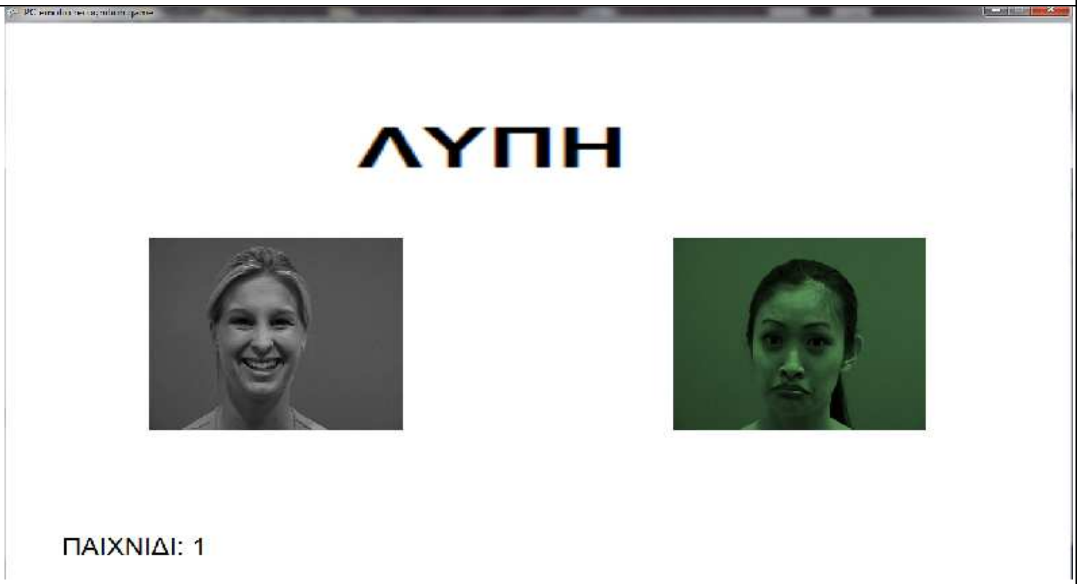
The reason why storyboarding is useful for us during software engineering is that it helps us to understand exactly how the software will work, much better than an abstract description. Storyboards not only help us to get a grip on target groups, context, product use and timing, but also in communicating about these aspects with all people involved.

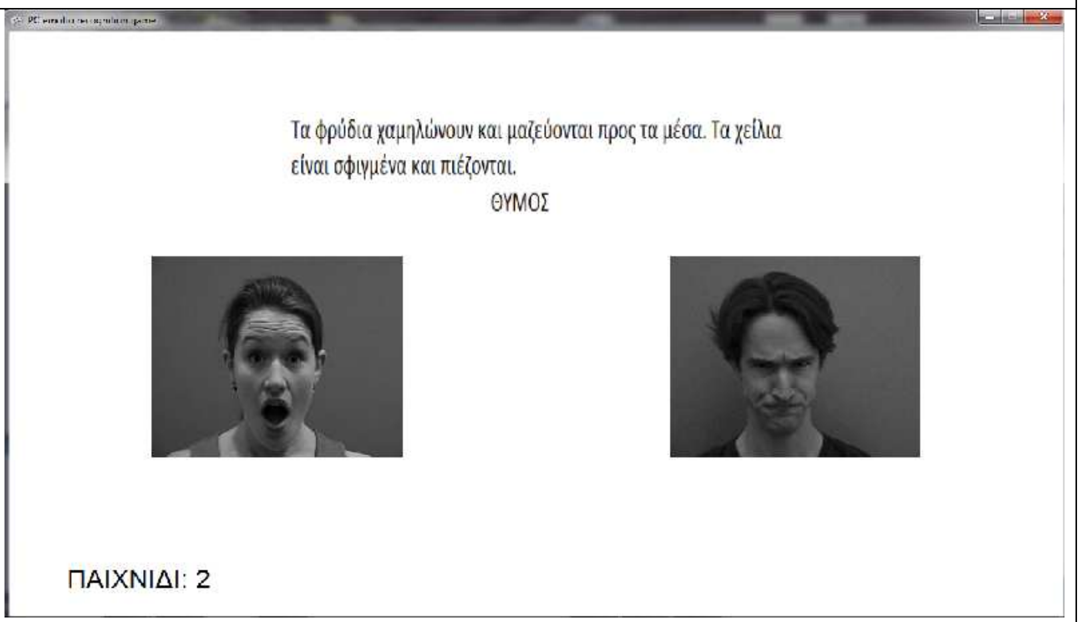
Our game's storyboard is a sequence of screenshots that show the levels of the game and the different scenes and goals.

SHOT	IMAGE
1	
DESCRIPTION	
Game begins with an instruction page where the child is informed what is going to happen, what he/she has to do and how he/she can do it. We select black context presented on a white background and grayscale images. A two-hand gesture which is performed by moving both hands above the head is required to start the game.	

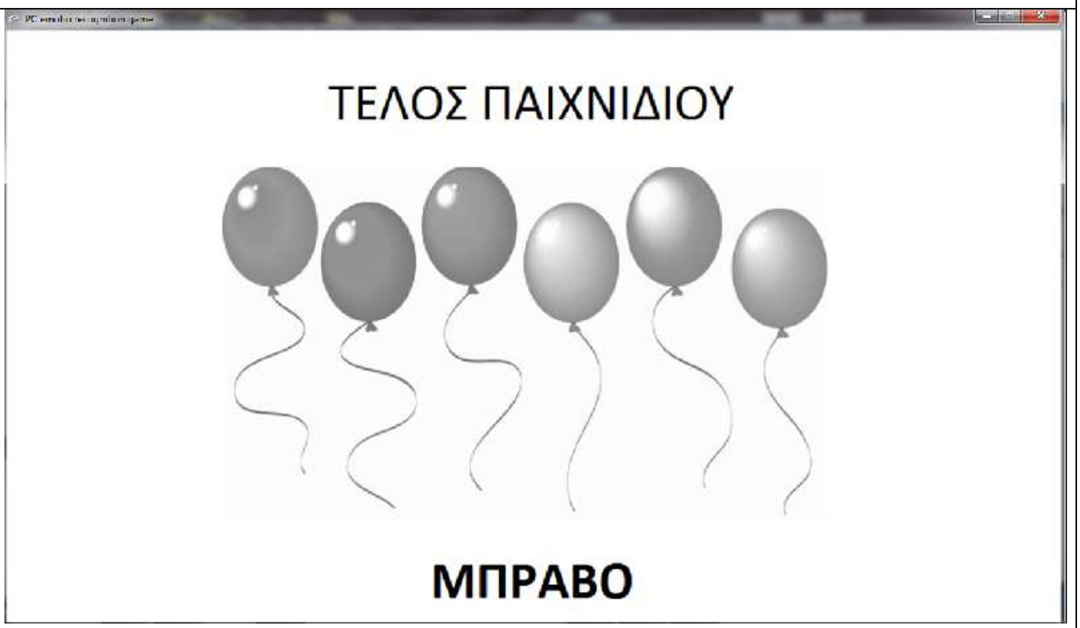
<table>
<tr><th>SHOT</th><th>IMAGE</th></tr>
<tr><td>2</td><td>
</td></tr>
<tr><th colspan="2">DESCRIPTION</th></tr>
<tr><td colspan="2">In the first level, children should learn labeling emotions by correlating emotion terms with images. The stimuli are presented on each trial with different pair of photos and the goal is to choose the correct image among the two.</td></tr>
</table>

<table>
<tr><th>SHOT</th><th>IMAGE</th></tr>
<tr><td>3</td><td>
</td></tr>
<tr><th colspan="2">DESCRIPTION</th></tr>
<tr><td colspan="2">Selecting the left image (the orientation of the image is decided by looking toward the screen) requires a one-hand gesture which is performed by moving the left hand above the head. Image’s color changes into light red for incorrect answer.</td></tr>
</table>

SHOT	IMAGE
4	
DESCRIPTION	
Selecting the right image (the orientation again of the image is decided by looking toward the screen) requires a one-hand gesture which is performed by moving the right hand above the head. Image's color changes into light green for the correct answer. Moving to the next play area requires a two-hand gesture which is performed by moving both hands above the head	

SHOT	IMAGE
5	
DESCRIPTION	
In the second level they should learn to recognize emotions from their description and their association with facial features.	

SHOT	IMAGE
6	
DESCRIPTION	
In the third level they should learn to identify the causes of various feelings in different situations, obtained through the use of social stories.	

SHOT	IMAGE
7	
DESCRIPTION	
“End of the game” screen.	

6.3 Gestures design

In order to create efficient natural user interfaces the idea is to design multiple gesture recognitions for the three tested commands; left, right and next. Once a gesture is operational, it can be tested to reveal its pros and cons. Then it is adapted, improved or dropped. The gestures of each command are put together in groups of three, one for each command. The interaction device that is used for this thesis work is Microsoft Kinect because it is a low cost and powerful NUI device.

6.3.1 Control mouse gesture

Our first idea was to simulate the mouse events with Kinect. We used a DLL file "KinectMouseController.DLL" in our C# project in order to control the cursor of Windows Operating System. In Windows the mouse_event function, synthesizes mouse motion and button click. User32 is a Microsoft DLL that provides a lower- level control of mouse and allows a control of Windows User Interface. In our project we added a reference to this DLL. Our program starts with the detection of the first skeleton. After that we get the right hand position (X and Y) of that skeleton and those coordinates are passed to a method as the mouse position. In this method we also pass screen dimensions and a Boolean that represents the mouse click (true = pressed, false=released). In our program, right hand moves the cursor and left hand performs the click (above the head means left mouse button pressed and below head means released).

Figure 32: Kinect as a mouse (left click)

The implementation was very complex and the results were not so good. It was very difficult to move the cursor in the lower part of the screen and pointing wasn't a nice and natural feeling. Mouse has been developed for pointing and click and it is obviously better than Kinect in this field. So, this idea was rejected.

Figure 33: Kinect as a mouse (pointing at the lower part of screen)

6.3.2 Complex gestures

The idea is to find natural gestures that would come intuitively to mind and also be ease to perform, to see if those gestures can be efficient in practice or not. The classic gestures of swipe were used as inspiration. For the left image selection, the orientation of the image is decided by looking toward the screen, required a two-hand gesture which is performed by moving the right hand above the head and swipe to right. For the right image selection, the orientation of the image is decided again by looking toward the screen, required also a two-hand gesture which is performed by moving the left hand above the head and swipe to left. For the moving to the next play area required again a two-hand gesture which is performed by moving both hands above the head.

The implementation is easy. It just computes the distance between joints. For the left gesture we check if the right hand's joint position is greater than head's joint position and if the left hand's joint position is greater than right hand's joint position.

```
if (leftHand.Position.X > rightHand.Position.X && rightHand.Position.Y>head.Position.Y)
```

Figure 34: Left gesture – both hands

For the right gesture we check if the left hand's joint position is greater than head's joint position and if the right hand's joint position is greater than left hand's joint position.

```
if (rightHand.Position.X <leftHand.Position.X && leftHand.Position.Y>head.Position.Y)
```

Figure 35: Right gesture – both hands

For the next gesture we check if both hands' (left and right) joint position is greater than head's joint position. The "Gesture" method takes of care of it.

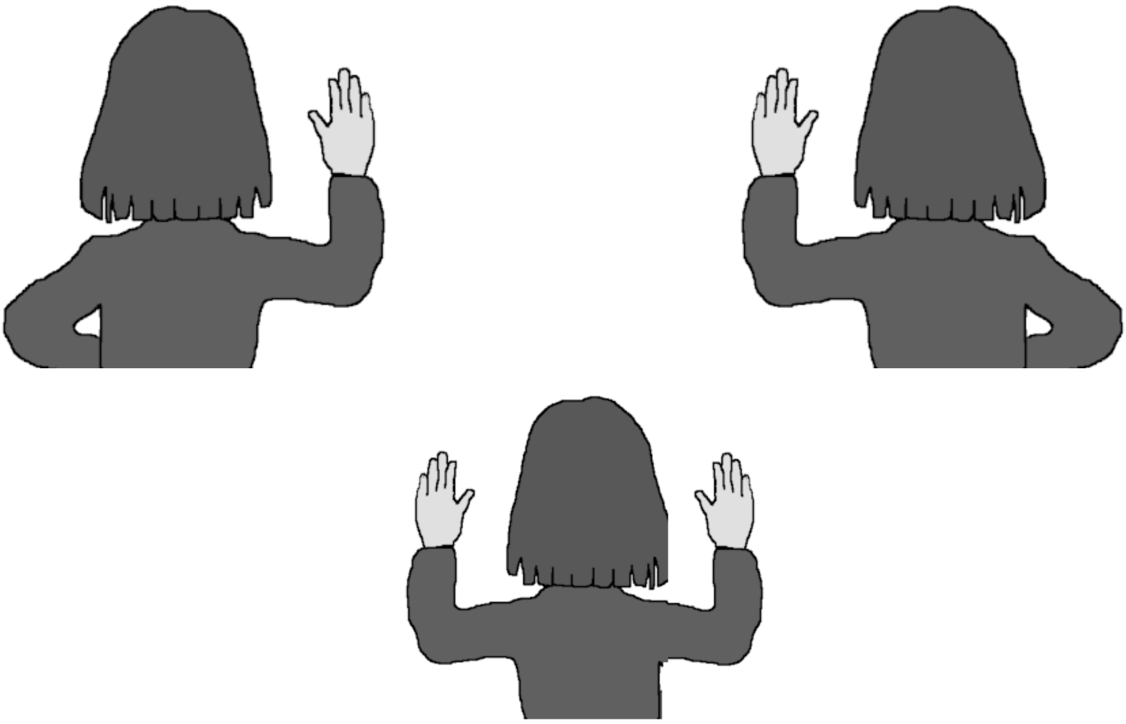
```
GamePC2.Game1                                   Gesture1()

private void Gesture()
{
    if (skeleton != null)
    {
        foreach (Joint joint in skeleton.Joints)
        {
            head = skeleton.Joints[JointType.Head];
            rightHand = skeleton.Joints[JointType.HandRight];
            leftHand = skeleton.Joints[JointType.HandLeft];

            if (leftHand.Position.Y > head.Position.Y && rightHand.Position.Y>head.Position.Y)
            {
                gameState = GameState.GamePlay;
            }
        }
    }
}
```

Figure 36: Next gesture

The Problems with this method are multiple. First it is difficult to explain with text or speech a young child how to perform this gesture. Also it may be difficult for children with autism who experience fine motor skills deficits to move their hands that way. For those reasons this solution was also rejected.

6.3.3 Simple gestures

Another approach for the left and right gesture selected for the game. The next gesture remained the same as before and we changed left gesture with the simple "lift you left hand above your head" and the right gesture with the "lift your right hand above your head". The implementation was easy again and is presented in the Implementation section.



7. Implementation

Our game is implemented with the use of Kinect, which is a motion sensing input device by Microsoft for Windows PCs. Natural Interaction is comprised of various middleware components that work in conjunction to translate real life actions, in to events or actions in a game. With Natural Interaction, the Kinect is able to track hand gestures, interpret body motions, and received vocal commands. Middleware is a component of the firmware that is used to accurately distinguish “human body parts and joints as well as distinguish individual human faces from one another”. From the RGBD information provided by the cameras, higher-end middleware can track the motion of the body, or even movement of an object. Natural Interaction can therefore precisely track movements in each part of the body, while “knowing” exactly who is using the device.

7.1 Hardware requirements

We implemented the game with the use of a Kinect sensor for Xbox and a PC with Microsoft Windows 7. To use a Kinect sensor and to be able to play our game, you will need a Kinect sensor (for Xbox or Windows) and a PC with the following:

- Windows 7 or Windows 8
- Windows 7, 32 bit (x86) or 64 bit (x64) processor
- Dual-core 2.66-GHz or faster processor
- Dedicated USB 2.0 bus
- 2 GB RAM

7.2 Software requirements

A cheap and simple way to do motion capturing is using the Microsoft Kinect. It offers simple and reliable skeleton tracking as well as a choice of both proprietary and open source SDKs.

In June 2011, Microsoft released a non-commercial Kinect_SDK for Windows that includes Windows 7-compatible PC drivers for the Kinect device (Microsoft’s SDK does not support older Windows versions or other operating systems).

Microsoft’s SDK allows developers to build Kinect enabled applications in Microsoft Visual Studio 2010 using C++, C# or Visual Basic. Microsoft is planning to release a commercial version of the Kinect for Windows SDK with support for more advanced device functionalities.

The simplicity of Kinect for Windows made it very easy to within a short time start to develop something of your own, making it a good tool for implementing our game. Since it was made to be used in the .NET environment it can also be used for developing games with XNA.

By using Visual studios we were able to use the Kinect SDK 1.6 released by Microsoft with C# as backend. The advantage of using C# was that we could integrate XNA Game Studio 4.0 to develop our game using Kinect. Writing the project’s software in C# provided another advantage. It would make it possible to utilize the XNA libraries (provided by Microsoft) to create the graphics.

7.2.1Kinect for Windows Software Development Toolkit

The Kinect for Windows SDK provides the tools and APIs, both native and managed, which needed to develop Kinect-enabled applications for Microsoft Windows. Developing Kinect-enabled applications is essentially the same as developing other Windows applications, except that the Kinect SDK provides support for the features of the Kinect, including color images, depth images, audio input, and skeletal data.

Some examples of the types of Windows applications that can be built using the functionality supported in this SDK are:

- Recognize and track moving people using skeletal tracking.
- Determine the distance between an object and the sensor camera using depth data.
- Capture audio using noise and echo cancellation or find the location of the audio source.

- Enable voice-activated applications by programming a grammar for use with a speech recognition engine.

The SDK includes drivers and technical documentation for implementing Kinect-enabled applications using a Kinect for Windows sensor and reference Application Programming Interfaces (APIs) and documentation for programming in managed and unmanaged code. Kinect SDK comes with NUI library which makes image streaming, depth streaming, audio streaming, 20 different skeleton joints and adjustable Kinect tilt reachable by the application.

7.2.2 XNA Game Studio

XNA Game Studio 4.0 is a programming environment that allows using Visual Studio to create games for Windows Phone, the Xbox 360 console, and Windows-based computers[6]. XNA Game Studio includes the XNA Framework, which is a set of managed libraries designed for game development based on Microsoft .NET Framework 2.0.

XNA Game Studio is an integrated development environment designed to make it easier to develop games. XNA Game Studio extends Microsoft Visual Studio with support for the XNA Game Studio Framework and tools. The XNA Game Studio Framework is a managed-code class library that contains features targeted specifically at game development. In addition, XNA Game Studio includes tools for adding graphic and audio content to your game.

The XNA Game Studio Framework is designed to follow the .NET Framework in its design patterns and idioms. With XNA Game Studio, we can use both the XNA Game Studio Framework and the more general .NET Framework for game development. The .NET Framework is a popular development platform for building apps for Windows, Windows Store, Windows Phone, Windows Server, and Windows Azure. The .NET Framework platform includes the C# and Visual Basic programming languages, the common language runtime, and an extensive class library.

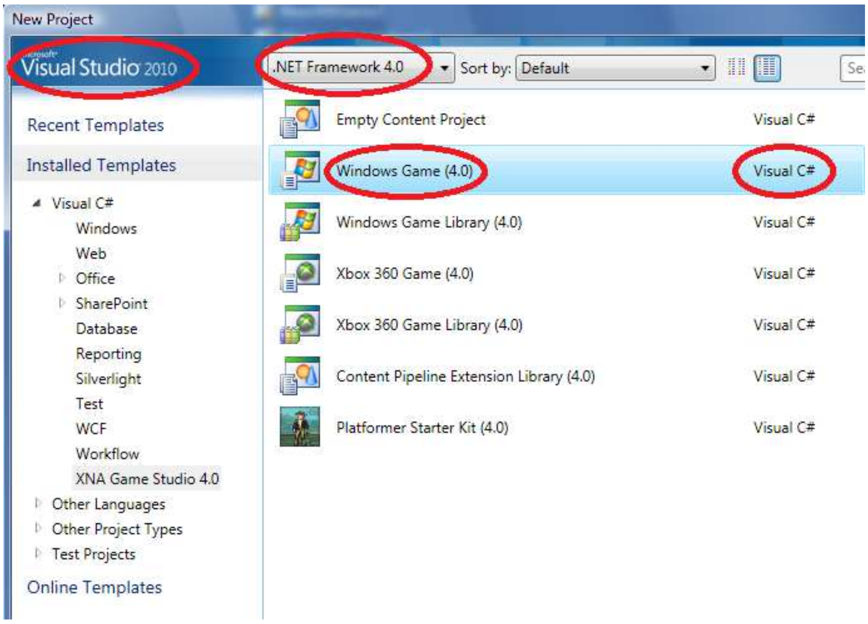


Figure 37: Software requirements

7.3 Interaction with the system

Interaction may be one of the areas that need to be developed with extreme care, depending on the activities and skills to be worked on each of the games and target group. The computer-based

[6] http://msdn.microsoft.com/en-us/library/bb200104(v=xnagamestudio.40).aspx

interventions that use a keyboard or a mouse for interaction might cause problem with the younger children which may not be able to use a computer. Our gesture-based interaction approach moves the control of computers from a mouse and keyboard, to the motions of the body via new input devices.

Our game is designed to be controlled by hand gestures. In our interactive system, the recognized gestures are translated into control commands. Player has three possible actions in all game states, to choose the left or the right image and to move to the next play area. These basic actions are implemented with efficient and easy to use gestures. Moving to the next play area requires a two-hand gesture which is performed by moving both hands above the head. Selecting the left image (the orientation of the image is decided by looking toward the screen) requires a one-hand gesture which is performed by moving the left hand above the head. Respectively, selecting the right image (the orientation again of the image is decided by looking toward the screen) requires a one-hand gesture which is performed by moving the right hand above the head.

During the game, if the player selects the correct or incorrect stimuli, the system will inform player that he/she gave the correct or incorrect answer. Each answer provides an audio and a visual feedback such as operation-related sounds and changing the images' color. A voice telling "Bravo" rewards player for the correct answer and a voice telling "Try again" encourages the player to try again when the user provides an incorrect answer. There are no other sound effects because individuals with ASD may suffer from auditory sensitivity [105], may demonstrate oversensitivity to certain sounds, even at low volume and may feel discomfort when exposed to certain sounds [106]. Visual feedback is also provided by changing the image's color into light green for the correct answer and into light red for incorrect answer. Light colors were selected because in ASD occur a reduced chromatic discrimination that is due to general reduction in sensitivity [107]. Children with autism are less accurate at detecting the differences between colors and have less accurate color perception [108]. Individuals with ASD are also experience hypersensitivity which includes increased light sensitivity and harshness of colors. Acute sensitivity to color is presented by their preference or avoidance of a particular color [109].

7.3.1 Gestures implementation

First of all we need to include the Kinect namespace, to add a Kinect sensor to our project and to initialize the sensor. Next we need to enable the depth stream. We chose stream resolution 640x480 at 30 frames per second. We get the depth data and we convert them to an image in order to display them on the screen. The depth stream returns a 16bit image, so we need to convert this to 32bit.

```
private byte[] ConvertDepthFrame(short[] depthFrame, DepthImageStream depthStream)
{
    int RedIndex = 0, GreenIndex = 1, BlueIndex = 2, AlphaIndex = 3;
    byte[] depthFrame32 = new byte[depthStream.FrameWidth * depthStream.FrameHeight * 4];
    for (int i16 = 0, i32 = 0; i16 < depthFrame.Length && i32 < depthFrame32.Length; i16++, i32 += 4)
    {
        int player = depthFrame[i16] & DepthImageFrame.PlayerIndexBitmask;
        int realDepth = depthFrame[i16] >> DepthImageFrame.PlayerIndexBitmaskWidth;
        // transform 13-bit depth information into an 8-bit intensity appropriate
        // for display (we disregard information in most significant bit)
        byte intensity = (byte)(~(realDepth >> 4));
        depthFrame32[i32 + RedIndex] = (byte)(intensity);
        depthFrame32[i32 + GreenIndex] = (byte)(intensity);
        depthFrame32[i32 + BlueIndex] = (byte)(intensity);
        depthFrame32[i32 + AlphaIndex] = 255;
    }
    return depthFrame32;
}
```

Figure 38: Convert depth data

We need a byte array to return our data in a 32bit format so we calculate its size based on the width and height of the depth frame image. Looping through the image we take each short of information and convert it to a byte with a bit shift right; we then apply the intensity to each of the channels (RGB) and make the alpha channel opaque. Finally we return our byte array containing the transformed pixel data.

We need to get the joints so we use an array to store the skeleton data from all skeletons and we then enable the skeleton stream. After enabling the skeleton stream we need to collect the skeleton data from the sensor. First we open the skeleton frame and test if it is null (no skeletons being tracked). If we have one or more skeletons that are tracked we need to get the data from the skeleton frame. First we check is the skeleton data array is empty or if the data from the previous frame has changed (more or less skeletons being tracked). If we meet one of these conditions we initialize our skeleton data array to the correct length. After taking the skeleton data from the frame we need to actually check the skeletons for their tracking state.

```
    using (SkeletonFrame skeletonFrame = imageFrames.OpenSkeletonFrame())
    {
        if (skeletonFrame != null)
        {
            if ((skeletonData == null) || (this.skeletonData.Length != skeletonFrame.SkeletonArrayLength))
            {
                this.skeletonData = new Skeleton[skeletonFrame.SkeletonArrayLength];
            }
            skeletonFrame.CopySkeletonDataTo(this.skeletonData);
        }
    }
    if (skeletonData != null)
    {
        foreach (Skeleton skel in skeletonData)
        {
            if (skel.TrackingState == SkeletonTrackingState.Tracked)
            {
                skeleton = skel;
            }
        }
    }
}
```

Figure 39: Skeleton data

With the skeleton data now available the gesture implementation is easy. Our program just computes the distance between joints and recognizes relaxed and discreet as well as large and demonstrative hand gestures. For the left gesture we check if the left hand joint position is greater than head joint position and if hip center join is greater than the right hand joint.


```
if (handLeft.Position.Y > head.Position.Y && handRight.Position.Y < hipCenter.Position.Y)
```

Figure 40: Left gesture

For the right gesture we check if the right hand joint position is greater than head joint position and the hip center joint is greater than the left hand joint.

```
if (handRight.Position.Y > head.Position.Y && handLeft.Position.Y < hipCenter.Position.Y)
```

Figure 41: Right gesture

For the next gesture we check if both hands (left and right) joint position is greater than head joint position.

```
if (leftHand.Position.Y > head.Position.Y && rightHand.Position.Y > head.Position.Y)
```

Figure 42: Next gesture

8. Discussion

Increased interest in the potential of technology for users with autism is motivated by the rapidly growing needs for providing intervention. Research shows that early intervention can greatly improve the lives of children with autism. Computer-based tools have been widely used with success to teach individuals with autism. However, early intervention cannot be achieved with computers due to lack of skills. Gesture-based interaction aims to contribute to overcoming this restriction.

Serious games with NUI interaction are a promising intervention strategy because they are appealing, motivating for young children to use and convenient to access. When the control is done through the natural gestures, the user does not have to learn how to perform the action and how to operate the game. Gesture-based interventions in order to help children with autism to improve their skills must be carefully designed in accordance with their abilities and needs.

8.1 Future work

The Kinect sensor offers an unlimited number of opportunities for new applications such as gesture-based interventions for individuals with autism. As Kinect sensor can support gesture and speech recognition and offer motion-sensing and interaction, children with autism can benefit to improve a broader range of skills.

A testing phase for the system is planned where we will attempt to validate the game design and to evaluate the game's usability. Future work could also include special designed gestures in order to improve their fine motor skills or design a Serious Game that will enable child to make facial emotion expressions that the system will be able to detect in order to provide a method called "learn by doing".

References

[1] "NIMH - National Institute of Mental Health," [Online]. Available: http://www.nimh.nih.gov/index.shtml. [Accessed 25 November 2012].

[2] American Psychiatric Association, *Diagnostic and Statistical Manual of Mental Disorders, Fourth Edition – Text Revision (DSM-IV-TR),* Washington, DC: American Psychiatric Association, 2000.

[3] World Health Organisation, *ICD-10 Classifications of Mental and Behavioural Disorder: Clinical Descriptions and Diagnostic Guidelines,* Geneva: World Health Organisation, 1992.

[4] P. Whiteley et al., "Gender Ratios in Autism, Asperger Syndrome and Autism Spectrum Disorder," *Autism Insights,* vol. 2, pp. 17-24, 05 March 2010.

[5] A. Westphal et al., "Revisiting Regression in Autism: Heller's Dementia Infantilis," *Journal of Autism and Developmental Disorders,* pp. 1-7, 08 June 2012.

[6] L. Kanner, "Autistic disturbances of affective contact," *The Nervous Child,* vol. 2, pp. 217-250, 1943.

[7] H. Asperger, "Die "Autistischen Psychopathen" im Kindesalter," *Archiv für Psychiatrie und Nervenkrankheiten,* vol. 117, pp. 76-136, 1944.

[8] U. Frith, Autism and Asperger syndrome, Cambridge: Cambridge University Press, 1991, pp. 37-92.

[9] J. N. Miller and S. Ozonoff, "Did Asperger's Cases Have Asperger Disorder? A Research Note," *Journal of Child Psychology and Psychiatry,* vol. 38, pp. 247-251, 1997.

[10] . A. D. Rasalam et al., "Characteristics of fetal anticonvulsant syndrome associated autistic disorder," *Developmental Medicine & Child Neurology,* vol. 47, no. 8, pp. 551-555, 2005.

[11] G. Williams et al., "Fetal valproate syndrome and autism: additional evidence of an association," *Developmental Medicine & Child Neurology,* vol. 43, pp. 202-206, 2001.

[12] S. Ozonoff et al., "Recurrence Risk for Autism Spectrum Disorders: A Baby Siblings Research Consortium Study," *Pediatrics,* vol. 128, no. 3, pp. e488-e495, 2011.

[13] S. E. Levy, et al., "Autism spectrum disorder and co-occurring developmental, psychiatric, and medical conditions among children in multiple populations of the United States," *Journal of Developmental and Behavioral Pediatrics,* vol. 31, no. 4, pp. 267-275, May 2010.

[14] M. S. Durkin, et al., "Advanced parental age and the risk of autism spectrum disorder,"

American Journal of Epidemiology, vol. 168, no. 11, pp. 1268-1276, 2008.

[15] D. Schendel and . T. K. Bhasin, "Birth weight and gestational age characteristics of children with autism, including a comparison with other developmental disabilities," *Pediatrics,* vol. 121, no. 6, p. 1155–1164, 2008.

[16] J. Baio, "Prevalence of autism spectrum disorders--Autism and Developmental Disabilities Monitoring Network, 14 sites, United States, 2008," *MMWR Surveillance Summaries,* vol. 61, no. 3, pp. 1-19, 30 March 2012.

[17] M. Elsabbagh et al., "Global Prevalence of Autism and Other Pervasive Developmental Disorders," *Autism Research,* vol. 5, no. 3, pp. 160-179, 11 April 2012.

[18] Y. S. Kim, et al., "Prevalence of autism spectrum disorders in a total population sample," *The American Journal of Psychiatry,* vol. 168, no. 9, pp. 904-912, 2011.

[19] "Ελληνική Εταιρία Προστασίας Αυτιστικών Ατόμων (Ε.Ε.Π.Α.Α.)," [Online]. Available: http://www.autismgreece.gr/. [Accessed 15 November 2012].

[20] A. Stampoltzis et al., "Developmental, familial and educational characteristics of a sample of children with Autism Spectrum Disorders in Greece," *Research in Autism Spectrum Disorders,* vol. 6, no. 4, pp. 1297-1303, 2012.

[21] C. P. Johnson and S. M. Myers, "Identification and Evaluation of Children With Autism Spectrum Disorders," *Pediatrics,* vol. 120, no. 5, pp. 1183-1215, 2007.

[22] D. L. Robins et al., "The Modified Checklist for Autism in Toddlers: An Initial Study Investigating the Early Detection of Autism and Pervasive Developmental Disorders," *Journal of Autism and Developmental Disorders,* vol. 31, no. 2, pp. 131-144, 2001.

[23] S. Skafidas et al., "Predicting the diagnosis of autism spectrum disorder using gene pathway analysis," *Molecular Psychiatry,* pp. 1-7, 2012.

[24] P. Ekman, "Facial Expressions," in *Dalgleish, T., & Power, M., Handbook of Cognition and Emotion*, New York, John Wiley & Sons Ltd, 1999, pp. 45-60.

[25] P. Ekman, "An argument for basic emotions," vol. 6, pp. 169-200, 1992.

[26] P. Ekman and W. V. Friesen, Facial Action Coding System, Palo Alto: Consulting Psychologist Press, 1978.

[27] M. H. Bornstein and M. E. Arterberry, "Recognition, discrimination and categorization of smiling by 5-month-old infants," *Developmental Science,* vol. 5, p. 585–599, 2003.

[28] G. Schwarzer and N. Zauner, "Face processing in 8-month-old infants: evidence for configural and analytical processing," *Vision Research,* vol. 43, no. 26, pp. 2783-2973, 2003.

[29] S. C. Widen and J. A. Russell, "Children acquire emotion categories gradually," *Cognitive Development,* vol. 23, no. 2, pp. 291-312, 2008.

[30] H. M. Wellmana, et al., "Early understanding of emotion: Evidence from natural language," *Cognition & Emotion,* vol. 9, no. 2-3, pp. 117-149, 1995.

[31] K. Durand, "The development of facial emotion recognition: The role of conWgural information," *Journal of Experimental Child Psychology,* vol. 97, pp. 14-27, 2007.

[32] S. W. White, K. Keonig and L. Scahill, "Social skills development in children with autism spectrum disorders: A review of the intervention research," *Journal of Autism and Developmental Disorders,* vol. 37, no. 10, pp. 1858-1868, 2007.

[33] P. R. Hobson, "The autistic child's appraisal of expressions of emotion: a further study," *Journal of Child Psychology and Psychiatry,* vol. 27, no. 5, p. 671–680, September 1986.

[34] K. M. Rump, et al., "The development of emotion recognition in individuals with autism," *Child Development,* vol. 80, no. 5, pp. 1434-1447, 2009.

[35] E. Hill, S. Berthoz and U. Frith, "Brief report: cognitive processing of own emotions in individuals with autistic spectrum disorder and in their relatives," vol. 34, no. 2, pp. 229-235, 2004.

[36] M. D. Lerner, J. C. McPartland and J. P. Morris, "Multimodal emotion processing in autism spectrum disorders: An event-related potential study," *Developmental Cognitive Neuroscience,* vol. 3, pp. 11-21, 2013.

[37] S. Kuusikko, et al., "Emotion recognition in children and adolescents with autism spectrum disorders," *Journal of Autism and Developmental Disorders,* vol. 39, no. 6, pp. 938-945, 2009.

[38] M. Uljarevic and A. Hamilton, "Recognition of Emotions in Autism: A Formal Meta-Analysis," *Journal of Autism and Developmental Disorders,* Vols. doi: 10.1007/s10803-012-1695-5, 2012.

[39] K. A. Pelphrey et al., "Visual Scanning of Faces in Autism," *Journal of Autism and Developmental Disorders,* vol. 32, no. 4, pp. 249-261, August 2002.

[40] W. Jones et al., "Absence of preferential looking to the eyes of approaching adults predicts level of social disability in 2-year-olds with autism spectrum disorder," *Arch Gen Psychiatry,* vol. 65, no. 8, pp. 946-954, 2008.

[41] M. B. Harms, A. Martin and G. L. Wallace, "Facial emotion recognition in autism spectrum disorders: A review of behavioral and neuroimaging studies," *Neuropsychology Review,* vol. 20, no. 3, pp. 290-322, 2010.

[42] J. W. Tanaka, et al., "The perception and identification of facial emotions in individuals with autism spectrum disorders using the Let's Face It! Emotion Skills Battery," *Journal of Child*

Psychology and Psychiatry, vol. 53, no. 12, pp. 1259-1267, 2012.

[43] J. L. Tracy, et al., "Is emotion recognition impaired in individuals with autism spectrum disorders?," vol. 41, no. 1, pp. 102-109, 2011.

[44] N. Bauminger, "The facilitation of social-emotional understanding and social interaction in high-functioning children with autism: Intervention outcomes," *Journal of Autism and Developmental Disorders,* vol. 32, no. 4, pp. 283-298, 2002.

[45] S. Faja, et al., "Becoming a face expert: A computerized face-training program for high-functioning individuals with autism spectrum disorders," *Developmental Neuropsychology,* vol. 33, no. 1, pp. 1-24, 2008.

[46] S. Eikeseth, "Outcome of comprehensive psycho-educational interventions for young children with autism," *Research in Developmental Disabilities,* vol. 30, no. 1, pp. 158-178, 2009.

[47] C. A. Gray and J. D. Garand, "Social stories: Improving responses of students with autism with accurate social information," *Focus on Autism and Other Developmental Disabilities,* vol. 8, no. 1, pp. 1-10, 1993.

[48] K. Pajareya and K. Nopmaneejumruslers, "A One-Year Prospective Follow-Up Study of a DIR/Floortime™ Parent Training Intervention for Preschool Children with Autistic Spectrum Disorders," *Journal of The Medical Association of Thailand,* vol. 95, no. 9, pp. 1184-1193.

[49] B. M. Prizant, et al. , "The SCERTS model: A transactional, family-centered approach to enhancing communication and socioemotional abilities of children with autism spectrum disorder," *Infants & Young Children,* vol. 4, pp. 296-316, 2003.

[50] S. Axelrod, K. K. McElrath and B. Wine, "Applied behavior analysis: autism and beyond," *Behavioral Interventions,* vol. 27, no. 1, pp. 1-15, 2012.

[51] B. Reichow, "Overview of meta-analyses on early intensive behavioral intervention for young children with autism spectrum disorders," *Journal of Autism and Developmental Disorders,* vol. 42, no. 4, pp. 512-520, 2012.

[52] P. Tzanakaki, et al., "How and why do parents choose early intensive behaivoral intervention for their young child with autism?," *Education and Training in Autism and Developmental Disabilities,* vol. 47, no. 1, pp. 58-71, 2012.

[53] S. J. Rogers, "Brief report: Early intervention in autism," *Journal of Autism and Developmental Disorders,* vol. 26, no. 2, pp. 243-246, 1996.

[54] A. C. Stahmer, N. Akshoomoff and A. B. Cunningham, "Inclusion for toddlers with autism spectrum disorders The first ten years of a community program," *Autism,* vol. 15, no. 5, pp. 625-641, 2011.

[55] T. Grandin, Thinking in pictures: And other reports from my life with autism, London: Bloomsbury Publishing Plc, 2006.

[56] K. Plaisted, M. O'Riordan and S. Baron-Cohen, "Enhanced visual search for a conjunctive target in autism: A research note," *Journal of Child Psychology and Psychiatry,* vol. 39, no. 5, pp. 777-783, 1998.

[57] M. A. O'Riordan, et al., "Superior visual search in autism.," *ournal of Experimental Psychology: Human Perception and Performance,* vol. 27, no. 3, pp. 719-730, 2001.

[58] J. B. Ganz, R. L. Simpson and E. M. Lund, "The Picture Exchange Communication System (PECS): A Promising Method for Improving Communication Skills of Learners with Autism Spectrum Disorders," *Education and Training in Autism and Developmental Disabilities,* vol. 47, no. 2, pp. 176-186, 2012.

[59] A. K. Moody, "Visual supports for promoting social skills in young children: a family perspective.," *Association for Childhood Education International,* vol. 88, no. 3, 2012.

[60] G. B. Mesibov, V. Shea and E. Schopler, The TEACCH Approach to Autism Spectrum Disorders., New York: Springer, 2004.

[61] T. R. Goldsmith and L. A. LeBlanc, "Use of technology in interventions for children with autism," *Journal of Early and Intensive Behavior Intervention,* vol. 1, no. 2, pp. 166-178, 2004.

[62] A. L. Wainer and B. R. Ingersoll, "The use of innovative computer technology for teaching social communication to individuals with autism spectrum disorders," *Research in Autism Spectrum Disorders,* vol. 5, no. 1, pp. 96-107, 2011.

[63] B. O. Ploog, et al., "Use of Computer-Assisted Technologies (CAT) to Enhance Social, Communicative, and Language Development in Children with Autism Spectrum Disorders," *Journal of Autism and Developmental Disorders,* pp. 1-22, 2012.

[64] M. Silver and P. Oakes, "Evaluation of a new computer intervention to teach people with autism or Asperger syndrome to recognize and predict emotions in others," *Autism,* vol. 5, no. 3, pp. 299-316, 2001.

[65] O. Golan and S. Baron-Cohen, "Systemizing empathy: Teaching adults with Asperger syndrome or high-functioning autism to recognize complex emotions using interactive multimedia," *Development and Psychopathology,* vol. 18, no. 2, pp. 591-617, 2006.

[66] O. Grynszpan, . J.-C. Martin and J. Nadel, "Multimedia interfaces for users with high functioning autism: An empirical investigation," *International Journal of Human-Computer Studies,* vol. 66, no. 8, pp. 628-639, 2008.

[67] S. L. Finkelstein, et al., "cMotion: A new game design to teach emotion recognition and programming logic to children using virtual humans," *IEEE Virtual Reality Conference,* pp. 249-

250, 2009.

[68] J. W. Tanaka, et al., "Using computerized games to teach face recognition skills to children with autism spectrum disorder: The Let's Face It! program," *Journal of Child Psychology and Psychiatry,* vol. 51, no. 8, pp. 944-952, 2010.

[69] D. Deriso, et al., "Emotion Mirror: A Novel Intervention for Autism Based on Real-Time Expression Recognition," vol. 7585, pp. 671-674, 2012.

[70] J. C. Miranda, et al., "Interactive Technology: Teaching People with Autism to Recognize Facial Emotions," *Autism Spectrum Disorders - From Genes to Environment,* 2011.

[71] R. el Kaliouby, A. Teeters and R. W. Picard, "An exploratory social-emotional prosthetic for autism spectrum disorders," *International Workshop on Wearable and Implantable Body Sensor Networks,* pp. 2-4, 2006.

[72] S. Baron-Cohen, et al., "Transported to a world of emotion," *The Psychologist,* vol. 20, no. 2, pp. 76-77, 2007.

[73] S. Baron-Cohen, O. Golan and E. Ashwin, "Can emotion recognition be taught to children with autism spectrum conditions?," *Philosophical Transactions of the Royal Society B-Biological Sciences,* vol. 364, no. 1535, pp. 3567-3574, 2009.

[74] O. Golan, et al., "Enhancing emotion recognition in children with autism spectrum conditions: an intervention using animated vehicles with real emotional faces," *Journal of Autism and Developmental Disorders,* vol. 40, no. 3, pp. 269-279, 2010.

[75] B. T. Williams, K. M. Gray and B. J. Tonge, "Teaching emotion recognition skills to young children with autism: a randomised controlled trial of an emotion training programme," *Journal of Child Psychology and Psychiatry,* vol. 53, no. 12, pp. 1268-1276, 2012.

[76] G. De Leo, et al., "A smart-phone application and a companion website for the improvement of the communication skills of children with autism: clinical rationale, technical development and preliminary results," *Journal of Medical Systems,* vol. 35, no. 4, pp. 703-711, 2011.

[77] E. Jasmin, et al., "Sensori-motor and daily living skills of preschool children with autism spectrum disorders," *Journal of Autism and Developmental Disorders,* vol. 39, no. 2, pp. 231-241, 2009.

[78] K. Sitdhisanguan, et al., "Using tangible user interfaces in computer-based training systems for low-functioning autistic children," *Personal and Ubiquitous Computing,* vol. 16, no. 2, pp. 143-155, 2012.

[79] A. E. Lane and J. M. Ziviani, "Factors influencing skilled use of the computer mouse by school-aged children," *Computers & Education,* vol. 55, no. 3, pp. 1112-1122, 2010.

[80] ISO, *ISO 9241-9 Ergonomic requirements for office work with visual display terminals (VDTs) - Part 9: Requirements for non-keyboard input devices,* International Organization for Standardization, 2000.

[81] D. Natapov, S. J. Castellucci and I. S. MacKenzie, "ISO 9241-9 evaluation of video game controllers," *Proceedings of Graphics Interface 2009,* pp. 223-230, May 2009.

[82] S. A. Douglas, A. E. Kirkpatrick and I. S. MacKenzie, "Testing pointing device performance and user assessment with the ISO 9241, Part 9 standard," *Proceedings of the SIGCHI conference on Human Factors in Computing Systems,* pp. 215-222, May 1999.

[83] I. S. MacKenzie, T. Kauppinen and M. Silfverberg, "Accuracy measures for evaluating computer pointing devices," *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,* pp. 9-16, 2001.

[84] M. Akamatsu and . I. S. MacKenzie, "Changes in applied force to a touchpad during pointing tasks," *International Journal of Industrial Ergonomics,* vol. 29, no. 3, pp. 171-182, 2002.

[85] D. Djaouti, et al., "Origins of Serious Games," *Serious Games and Edutainment Applications,* pp. 25-43, 2011.

[86] M. Zyda, "From visual simulation to virtual reality to games," *IEEE Computer Society,* vol. 38, no. 9, pp. 25-32, 2005.

[87] M. Noor, et al., "Serious Game for Autism Children: Review of Literature," *World Academy of Science, Engineering and Technology,* vol. 64, no. 124, pp. 647-652.

[88] R. A. Clark, et al., "Validity of the Microsoft Kinect for assessment of postural control," *Gait & Posture,* vol. 36, no. 3, pp. 372-377, 2012.

[89] A. D. Wilson, "Using a depth camera as a touch sensor," *ACM International Conference on Interactive Tabletops and Surfaces,* pp. 69-72, 2010.

[90] Z. Zhang , "Microsoft Kinect Sensor and Its Effect," *IEEE MultiMedia,* vol. 19, no. 2, pp. 4-10, 2012.

[91] C. Schonauer, et al., "Chronic pain rehabilitation with a serious game using multimodal input," *International Conference on Virtual Rehabilitation (ICVR),* pp. 1-8, 2011.

[92] K. F. Li, et al. , "A web-based sign language translator using 3d video processing," *14th International Conference on Network-Based Information Systems (NBiS),* pp. 356-361, 2011.

[93] L. Gallo. , A. P. Placitelli and M. Ciampi, "Controller-free exploration of medical image data: experiencing the Kinect," *24th International Symposium on Computer-Based Medical Systems (CBMS),* pp. 1-6, 2011.

[94] M. Van den Bergh, et al., "Real-time 3D hand gesture interaction with a robot for understanding directions from humans," *IEEE RO-MAN,* pp. 357-362, 2011.

[95] P. Benavidez and M. Jamshidi, "Mobile robot navigation and target tracking system," *6th International Conference on System of Systems Engineering (SoSE),* pp. 299-304, 2011.

[96] H. Du, et al., "Interactive 3D modeling of indoor environments with a consumer depth camera," *Proceedings of the 13th international conference on Ubiquitous computing,* pp. 75-84, 2011.

[97] J. Lozano-Martínez, F. J. Ballesta-Pagán and S. Alcaraz-García, "Software for Teaching Emotions to Students with Autism Spectrum Disorder," *Revista Comunicar,* vol. 18, no. 36, pp. 139-148, 2011.

[98] K. Mitgutsch and N. Alvarado, "Purposeful by design?: a serious game design assessment framework," *Proceedings of the International Conference on the Foundations of Digital Games,* pp. 121-128, 2012.

[99] B. Abirached, Y. Zhang and J. H. Park, "Understanding User Needs for Serious Games for Teaching Children with Autism Spectrum Disorders Emotions," *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications,* pp. 1054-1063, 2012.

[100] J. H. Park, B. Abirached and Y. Zhang, "A framework for designing assistive technologies for teaching children with ASDs emotions," *Extended Abstracts on Human Factors in Computing Systems,* pp. 2423-2428, 2012.

[101] E. Ashwin, et al., "Eagle-eyed visual acuity: an experimental investigation of enhanced perception in autism," *Biological Psychiatry,* vol. 65, no. 1, pp. 17-21, 2009.

[102] S. Dodd, Understanding Autism, Sydney: Elsevier, 2004.

[103] M. N. Dailey , G. W. Cottrell and J. Reilly, "CAlifornia Facial Expressions (CAFE)," 2001. [Online]. Available: http://www.cs.ucsd.edu/users/gary/CAFE/.

[104] M. L. Smith, et al., "Transmitting and decoding facial expressions," *Psychological Science,* vol. 16, no. 3, pp. 184-189, 2005.

[105] E. Gomes, F. S. Pedroso and M. B. Wagner, "Auditory hypersensitivity in the autistic spectrum disorder (original title: Hipersensibilidade auditiva no transtorno do espectro autístico)," *Pró-Fono Revista de Atualização Científica,* vol. 20, no. 4, pp. 279-284, 2008.

[106] Y. H. Tan, et al., "Auditory abnormalities in children with autism," *Open Journal of Psychiatry,* vol. 2, no. 1, pp. 33-37, 2012.

[107] A. Franklin, et al., "Reduced chromatic discrimination in children with autism spectrum

disorders," *Developmental Science,* vol. 13, no. 1, pp. 188-200, 2010.

[108] A. Franklin, et al., "Color perception in children with autism," *Journal of Autism and Developmental Disorders,* vol. 38, no. 10, pp. 1837-1847, 2008.

[109] R. A. Coulter, "Understanding the Visual Symptoms of Individuals with Autism Spectrum Disorder (ASD)," *Optometry & Vision Development,* vol. 40, no. 3, pp. 164-175, 2009.

Appendix

1. Software

- Microsoft Windows 7
- XNA Game Studio 4.0
  http://www.microsoft.com/en-us/download/details.aspx?id=23714

  Instructions
1. Install Microsoft Visual Studio 2010.
2. Obtain the latest updates for Visual Studio from Microsoft Update.
3. Download and run the Microsoft XNA Game Studio 4.0 installer.
4. Follow the setup instructions.
5. Launch Visual Studio 2010 from Microsoft XNA Game Studio 4.0 on the Start menu.

- Microsoft .NET Framework
- Kinect for Windows SDK and Toolkit
  http://www.microsoft.com/en-us/kinectforwindows/develop/developer-downloads.aspx

  To Install the SDK and Toolkit
1. Make sure the Kinect sensor is not plugged into the USB port on the computer.
2. If you have Kinect for Windows SDK 1.0 installed, close any open samples, the Sample Browser, etc. You do not need to uninstall SDK 1.0. Skip to step 5.
3. Remove any other drivers for the Kinect sensor.
4. If you have Microsoft Server Speech Platform 10.2 installed, uninstall the Microsoft Server Speech Platform Runtime and SDK components including both the x86 and x64 bit versions, plus the Microsoft Server Speech Recognition Language – Kinect language pack.
5. Close Visual Studio. You must close Visual Studio before installing the SDK and then restart it after installation to pick up the KINECTSDK10_DIR environment variables that the SDK requires.
6. From the download location, double-click the KinectSDK-*versionNumber*-Setup.exe (where version number is the latest shipped version). This single installer works for both 32-bit and 64-bit versions of Windows.
7. Once this SDK has completed installing successfully, ensure the Kinect sensor is plugged into an external power source then plug the sensor into the PC's USB port. The drivers will load automatically.
8. The Kinect sensor should now be working correctly.
9. To install the Toolkit, from the download location double-click on KinectDeveloperToolkit-*versionNumber*-Setup.exe (where version number is the latest shipped version). This single installer works for both 32-bit and 64-bit Windows.
10. To execute tools or samples, install samples, or explore other resources, start the Developer Toolkit Browser via the start menu: **Kinect for Windows SDK *version number* > Developer Toolkit Browser *version number* (Kinect for Windows)**.

  If the driver is installed correctly:
1. The solid green LED on the Kinect sensor should be lit.
2. The Kinect sensor should appear in **Device Manager** as the following nodes under **Kinect for Windows**: **Kinect for Windows Audio Array Control**, **Kinect for Windows Camera**, and **Kinect for Windows Security Control**.
3. The Kinect sensor's microphone array will appear under **Sound, video and game controllers** in **Device Manager** as: **Microsoft Kinect USB Audio**.

2. Code

```
using System;
using System.Collections.Generic;
using System.Linq;
using Microsoft.Xna.Framework;
using Microsoft.Xna.Framework.Audio;
using Microsoft.Xna.Framework.Content;
using Microsoft.Xna.Framework.GamerServices;
using Microsoft.Xna.Framework.Graphics;
using Microsoft.Xna.Framework.Input;
using Microsoft.Xna.Framework.Media;


//kinect
using Microsoft.Kinect;

namespace emotionGame
{
    /// <summary>
    /// This is the main type for your game
    /// </summary>
    public class Game1 : Microsoft.Xna.Framework.Game
    {
        GraphicsDeviceManager graphics;
        SpriteBatch spriteBatch;

//kinect ---> arxi

        //add kinect sensor
        KinectSensor kinect;

        //place to store skeleton data
        Skeleton[] skeletonData;

        //place to store our tracked skeleton
        Skeleton skeleton;

        //a texture to store our Depth Stream
        Texture2D depthVideo, jointTexture;

        // textures for the joints
        Joint rightHand, leftHand, head;

//kinect ----> telos


//Grafika kai paixnidi

        //gia to megethos tou parathirou
        int w, h;

        int i = 0;


      //   Boolean game = false;

        enum GameState
        {
            StartMenu,
            GamePlay,
            GamePlay1,
```

```
        GamePlay2,
        GamePlay3,
        GamePlay4,
        GamePlay5,
        GamePlay6,
        GamePlay7,
        GamePlay8, GamePlay9, GamePlay10,GamePlay11, GamePlay12,
        GamePlay13, GamePlay14, GamePlay15, GamePlay16,GamePlay17,
        GamePlay18, GamePlay19, GamePlay20,GamePlay21, GamePlay22,
        GamePlay23, GamePlay24, GamePlay25, GamePlay26, GamePlay27,
        GamePlay28, GamePlay29, GamePlay30
    }

    GameState gameState;


    // thesi gia ti katastasi tou pliktrologiou
    KeyboardState myKeyboard;

    //Gia tis odigies kai gia to titlo kai tis eikones
    Vector2 instrPos, titlePos, textPos, leftImgPos, rightImgPos, lastPos, gamePos;
    Rectangle instrRec, titleRec, textRec, leftImgRec, rightImgRec, lastRec, gameRec;
    Texture2D instr, title, last;

    //pinakes pou tha valo tis lekseis kai ta emotions
    Texture2D[] text = new Texture2D[15];
    Texture2D[] emotion = new Texture2D[30];
    Texture2D[] game = new Texture2D[3];

    Song[] song = new Song[15];
    Song bravo, odigies, pali, telos, titlos;

    //xroma eikonon
    Color leftImgColor = Color.White;
    Color rightImgColor = Color.White;


    //Gia na grapso keimeno
   //SpriteFont font;


    public Game1()
    {
        graphics = new GraphicsDeviceManager(this);
        Content.RootDirectory = "Content";



// grafika kai paixnidi

        // orizo to megethos tou parathirou
        graphics.PreferredBackBufferWidth = 1260;
        graphics.PreferredBackBufferHeight = 760;

    }

    /// <summary>
    /// Allows the game to perform any initialization it needs to before starting to run.
    /// This is where it can query for any required services and load any non-graphic
    /// related content.  Calling base.Initialize will enumerate through any components
    /// and initialize them as well.
    /// </summary>
    protected override void Initialize()
```

```
        {
            // TODO: Add your initialization logic here

    //kinect----> arxi

            //returns the first Kinect Sensor from the collection of Kinect Sensor
            kinect = KinectSensor.KinectSensors[0];

            //enable the color stream
            kinect.ColorStream.Enable(ColorImageFormat.RgbResolution640x480Fps30);

            //enable the depth stream
            kinect.DepthStream.Enable(DepthImageFormat.Resolution640x480Fps30);

            //enables skeleton stream
            kinect.SkeletonStream.Enable();

            //The color and depth frames are sent to the event handler at the same time so there is no lag between the
images
            kinect.AllFramesReady += new EventHandler<AllFramesReadyEventArgs>(kinect_AllFramesReady);

            //start the Kinect Sensor
            kinect.Start();

            //kinect tilt
            kinect.ElevationAngle = 5;

            //initialise the video texture before it is drawn
            depthVideo = new Texture2D(graphics.GraphicsDevice, kinect.DepthStream.FrameWidth,
kinect.DepthStream.FrameHeight);

    // kinect ---->telos


    //grafika kai paixnidi


            //Vazo title sto parathiro
            Window.Title = "PC emotio recognition game";

            // Vrisko to ipsos kai to platos tou parathirou
            w = Window.ClientBounds.Width;
            h = Window.ClientBounds.Height;


            //Thesi pou tha topothetisei ta antikeimena
            titlePos = new Vector2((w / 2) - 240, 20);
            instrPos = new Vector2(20, 100);
            textPos = new Vector2((w / 2) - 300, 90);
            leftImgPos = new Vector2(170, (h / 2) - 100);
            rightImgPos = new Vector2(790, (h / 2) - 100);
            lastPos = new Vector2((w / 2) - 400, (h / 2) - 340);
            gamePos = new Vector2(30, 630);


            //dimiourgia rectangle guro apo ta antikeimena
            titleRec = new Rectangle((int)titlePos.X, (int)titlePos.Y, 480, 60);
            instrRec = new Rectangle((int)instrPos.X, (int)instrPos.Y, 1220, 600);
            textRec = new Rectangle((int)textPos.X, (int)textPos.Y, 600, 150);
            leftImgRec = new Rectangle((int)leftImgPos.X, (int)leftImgPos.Y, 300, 250);
            rightImgRec = new Rectangle((int)rightImgPos.X, (int)rightImgPos.Y, 300, 250);
            lastRec = new Rectangle((int)lastPos.X, (int)lastPos.Y, 800, 680);
            gameRec = new Rectangle((int)gamePos.X, (int)gamePos.Y, 250, 100);
```

```
            base.Initialize();
        }
        /// <summary>
        /// LoadContent will be called once per game and is the place to load
        /// all of your content.
        /// </summary>
        protected override void LoadContent()
        {
            // Create a new SpriteBatch, which can be used to draw textures.
            spriteBatch = new SpriteBatch(GraphicsDevice);

// kinect ----> arxi

            //load joint.png
            jointTexture = Content.Load<Texture2D>("joint");

// kinect ----> telos


// grafika kai paixnidi


            //fortono tis eikones

            emotion[0] = Content.Load<Texture2D>("cafe/happy");
            emotion[1] = Content.Load<Texture2D>("cafe/sad");
            emotion[2] = Content.Load<Texture2D>("cafe/fear");
            emotion[3] = Content.Load<Texture2D>("cafe/angry");
            emotion[4] = Content.Load<Texture2D>("cafe/happy");
            emotion[5] = Content.Load<Texture2D>("cafe/surprised2");
            emotion[6] = Content.Load<Texture2D>("cafe/sad");
            emotion[7] = Content.Load<Texture2D>("cafe/fear");
            emotion[8] = Content.Load<Texture2D>("cafe/surprised2");
            emotion[9] = Content.Load<Texture2D>("cafe/happy");
            emotion[10] = Content.Load<Texture2D>("cafe/happy2");
            emotion[11] = Content.Load<Texture2D>("cafe/fear2");
            emotion[12] = Content.Load<Texture2D>("cafe/surprised");
            emotion[13] = Content.Load<Texture2D>("cafe/angry1");
            emotion[14] = Content.Load<Texture2D>("cafe/sad1");
            emotion[15] = Content.Load<Texture2D>("cafe/happy2");
            emotion[16] = Content.Load<Texture2D>("cafe/angry1");
            emotion[17] = Content.Load<Texture2D>("cafe/fear2");
            emotion[18] = Content.Load<Texture2D>("cafe/sad1");
            emotion[19] = Content.Load<Texture2D>("cafe/surprised");
            emotion[20] = Content.Load<Texture2D>("cafe/surprised1");
            emotion[21] = Content.Load<Texture2D>("cafe/fear1");
            emotion[22] = Content.Load<Texture2D>("cafe/sad2");
            emotion[23] = Content.Load<Texture2D>("cafe/surprised1");
            emotion[24] = Content.Load<Texture2D>("cafe/angry2");
            emotion[25] = Content.Load<Texture2D>("cafe/happy1");
            emotion[26] = Content.Load<Texture2D>("cafe/fear1");
            emotion[27] = Content.Load<Texture2D>("cafe/surprised1");
            emotion[28] = Content.Load<Texture2D>("cafe/happy1");
            emotion[29] = Content.Load<Texture2D>("cafe/angry2");



            text[0] = Content.Load<Texture2D>("texts/lupi");
            text[1] = Content.Load<Texture2D>("texts/thimos");
            text[2] = Content.Load<Texture2D>("texts/xara");
            text[3] = Content.Load<Texture2D>("texts/fovos");
```

```
    text[4] = Content.Load<Texture2D>("texts/ekpliksi");
    text[5] = Content.Load<Texture2D>("images/xara1");
    text[6] = Content.Load<Texture2D>("images/thimos1");
    text[7] = Content.Load<Texture2D>("images/lupi1");
    text[8] = Content.Load<Texture2D>("images/fovos1");
    text[9] = Content.Load<Texture2D>("images/ekpliksi1");
    text[10] = Content.Load<Texture2D>("stories/fovos2");
    text[11] = Content.Load<Texture2D>("stories/lupi2");
    text[12] = Content.Load<Texture2D>("stories/thimos2");
    text[13] = Content.Load<Texture2D>("stories/ekpliksi2");
    text[14] = Content.Load<Texture2D>("stories/xara2");

    game[0] = Content.Load<Texture2D>("texts/game1");
    game[1] = Content.Load<Texture2D>("texts/game2");
    game[2] = Content.Load<Texture2D>("texts/game3");

    instr = Content.Load<Texture2D>("texts/odigies");

    title = Content.Load<Texture2D>("texts/title");

    last = Content.Load<Texture2D>("texts/last");

    song[0] = Content.Load<Song>("sounds/xara");
    song[1] = Content.Load<Song>("sounds/xara1");
    song[2] = Content.Load<Song>("sounds/xara2");
    song[3] = Content.Load<Song>("sounds/lupi");
    song[4] = Content.Load<Song>("sounds/lupi1");
    song[5] = Content.Load<Song>("sounds/lupi2");
    song[6] = Content.Load<Song>("sounds/thimos");
    song[7] = Content.Load<Song>("sounds/thimos1");
    song[8] = Content.Load<Song>("sounds/thimos2");
    song[9] = Content.Load<Song>("sounds/fovos");
    song[10] = Content.Load<Song>("sounds/fovos1");
    song[11] = Content.Load<Song>("sounds/fovos2");
    song[12] = Content.Load<Song>("sounds/ekpliksi");
    song[13] = Content.Load<Song>("sounds/ekpliksi1");
    song[14] = Content.Load<Song>("sounds/ekpliksi2");

    bravo = Content.Load<Song>("sounds/bravo");
    odigies = Content.Load<Song>("sounds/odigies");
    pali = Content.Load<Song>("sounds/pali");
    telos = Content.Load<Song>("sounds/telos");
    titlos = Content.Load<Song>("sounds/title");


    // TODO: use this.Content to load your game content here
}

/// <summary>
/// UnloadContent will be called once per game and is the place to unload
/// all content.
/// </summary>
protected override void UnloadContent()
{
    // TODO: Unload any non ContentManager content here
}

/// <summary>
/// Allows the game to run logic such as updating the world,
/// checking for collisions, gathering input, and playing audio.
/// </summary>
/// <param name="gameTime">Provides a snapshot of timing values.</param>
protected override void Update(GameTime gameTime)
```

```
        {
            // Allows the game to exit
            if (GamePad.GetState(PlayerIndex.One).Buttons.Back == ButtonState.Pressed)
                this.Exit();


    // grafika kai paixnidi

            // Eksodos ap' to pexnidi otan patiso Escape. Kleisimo parathirou
            myKeyboard = Keyboard.GetState();
            if (myKeyboard.IsKeyDown(Keys.Escape))
                this.Exit();


            if (gameState == GameState.StartMenu)
            {
                Gesture();
                //MediaPlayer.Play(titlos);
                //MediaPlayer.Play(odigies);
            }

            if (gameState == GameState.GamePlay)
            {

                Gesture1(skeleton.Joints[JointType.Head],                                skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }

            if (gameState == GameState.GamePlay1)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay2)
            {
                if (i == 0)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture1(skeleton.Joints[JointType.Head],                                skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }
            if (gameState == GameState.GamePlay3)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay4)
            {
                if (i == 1)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture2(skeleton.Joints[JointType.Head],                                skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }

            if (gameState == GameState.GamePlay5)
            {
```

```
                Gesture();
            }

            if (gameState == GameState.GamePlay6)
            {
                if (i == 2)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture1(skeleton.Joints[JointType.Head],                         skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }


            if (gameState == GameState.GamePlay7)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay8)
            {
                if (i == 3)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture2(skeleton.Joints[JointType.Head],                         skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }

            if (gameState == GameState.GamePlay9)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay10)
            {
                if (i == 4)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture2(skeleton.Joints[JointType.Head],                         skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }


            if (gameState == GameState.GamePlay11)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay12)
            {
                if (i == 5)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
```

```
                    i = i + 1;
                }
                Gesture1(skeleton.Joints[JointType.Head],                          skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }


            if (gameState == GameState.GamePlay13)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay14)
            {
                if (i == 6)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture2(skeleton.Joints[JointType.Head],                          skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }


            if (gameState == GameState.GamePlay15)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay16)
            {
                if (i == 7)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture1(skeleton.Joints[JointType.Head],                          skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }
            if (gameState == GameState.GamePlay17)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay18)
            {
                if (i == 8)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture1(skeleton.Joints[JointType.Head],                          skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }

            if (gameState == GameState.GamePlay19)
            {
                Gesture();
            }
```

```
            if (gameState == GameState.GamePlay20)
            {
                if (i == 9)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture1(skeleton.Joints[JointType.Head],                    skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }

            if (gameState == GameState.GamePlay21)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay22)
            {
                if (i == 10)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture2(skeleton.Joints[JointType.Head],                    skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }


            if (gameState == GameState.GamePlay23)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay24)
            {
                if (i == 11)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture2(skeleton.Joints[JointType.Head],                    skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }


            if (gameState == GameState.GamePlay25)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay26)
            {
                if (i == 12)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture1(skeleton.Joints[JointType.Head],                    skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
```

```
            }

            if (gameState == GameState.GamePlay27)
            {
                Gesture();
            }

            if (gameState == GameState.GamePlay28)
            {
                if (i == 13)
                {
                    leftImgColor = Color.White;
                    rightImgColor = Color.White;
                    i = i + 1;
                }
                Gesture2(skeleton.Joints[JointType.Head],                skeleton.Joints[JointType.HandLeft],
skeleton.Joints[JointType.HandRight], skeleton.Joints[JointType.HipCenter]);
            }

            if (gameState == GameState.GamePlay29)
            {
                Gesture();
            }

            // TODO: Add your update logic here

            base.Update(gameTime);
        }

        /// <summary>
        /// This is called when the game should draw itself.
        /// </summary>
        /// <param name="gameTime">Provides a snapshot of timing values.</param>
        protected override void Draw(GameTime gameTime)
        {
            GraphicsDevice.Clear(Color.White);

            spriteBatch.Begin();


    // grafika kai paixnidi


            if (gameState == GameState.StartMenu)
            {
                spriteBatch.Draw(instr, instrRec, Color.White);
                spriteBatch.Draw(title,titleRec,Color.White);
            }

            if (gameState == GameState.GamePlay || gameState == GameState.GamePlay1)
            {
                spriteBatch.Draw(emotion[0], leftImgRec, leftImgColor);
                spriteBatch.Draw(emotion[1], rightImgRec, rightImgColor);
                spriteBatch.Draw(text[0], textRec, Color.White);
                spriteBatch.Draw(game[0], gameRec, Color.White);
            }


            if (gameState == GameState.GamePlay2 || gameState == GameState.GamePlay3)
            {
                spriteBatch.Draw(emotion[2], leftImgRec, leftImgColor);
                spriteBatch.Draw(emotion[3], rightImgRec, rightImgColor);
                spriteBatch.Draw(text[1], textRec, Color.White);
```

```
    spriteBatch.Draw(game[0], gameRec, Color.White);
}

if (gameState == GameState.GamePlay4 || gameState == GameState.GamePlay5)
{
    spriteBatch.Draw(emotion[4], leftImgRec, leftImgColor);
    spriteBatch.Draw(emotion[5], rightImgRec, rightImgColor);
    spriteBatch.Draw(text[2], textRec, Color.White);
    spriteBatch.Draw(game[0], gameRec, Color.White);
}

if (gameState == GameState.GamePlay6 || gameState == GameState.GamePlay7)
{
    spriteBatch.Draw(emotion[6], leftImgRec, leftImgColor);
    spriteBatch.Draw(emotion[7], rightImgRec, rightImgColor);
    spriteBatch.Draw(text[3], textRec, Color.White);
    spriteBatch.Draw(game[0], gameRec, Color.White);
}

if (gameState == GameState.GamePlay8 || gameState == GameState.GamePlay9)
{
    spriteBatch.Draw(emotion[8], leftImgRec, leftImgColor);
    spriteBatch.Draw(emotion[9], rightImgRec, rightImgColor);
    spriteBatch.Draw(text[4], textRec, Color.White);
    spriteBatch.Draw(game[0], gameRec, Color.White);
}


if (gameState == GameState.GamePlay10 || gameState == GameState.GamePlay11)
{
    spriteBatch.Draw(emotion[10], leftImgRec, leftImgColor);
    spriteBatch.Draw(emotion[11], rightImgRec, rightImgColor);
    spriteBatch.Draw(text[5], textRec, Color.White);
    spriteBatch.Draw(game[1], gameRec, Color.White);
}


if (gameState == GameState.GamePlay12 || gameState == GameState.GamePlay13)
{
    spriteBatch.Draw(emotion[12], leftImgRec, leftImgColor);
    spriteBatch.Draw(emotion[13], rightImgRec, rightImgColor);
    spriteBatch.Draw(text[6], textRec, Color.White);
    spriteBatch.Draw(game[1], gameRec, Color.White);
}

if (gameState == GameState.GamePlay14 || gameState == GameState.GamePlay15)
{
    spriteBatch.Draw(emotion[14], leftImgRec, leftImgColor);
    spriteBatch.Draw(emotion[15], rightImgRec, rightImgColor);
    spriteBatch.Draw(text[7], textRec, Color.White);
    spriteBatch.Draw(game[1], gameRec, Color.White);
}

if (gameState == GameState.GamePlay16 || gameState == GameState.GamePlay17)
{
    spriteBatch.Draw(emotion[16], leftImgRec, leftImgColor);
    spriteBatch.Draw(emotion[17], rightImgRec, rightImgColor);
    spriteBatch.Draw(text[8], textRec, Color.White);
    spriteBatch.Draw(game[1], gameRec, Color.White);
}

if (gameState == GameState.GamePlay18 || gameState == GameState.GamePlay19)
{
```

```
                spriteBatch.Draw(emotion[18], leftImgRec, leftImgColor);
                spriteBatch.Draw(emotion[19], rightImgRec, rightImgColor);
                spriteBatch.Draw(text[9], textRec, Color.White);
                spriteBatch.Draw(game[1], gameRec, Color.White);
            }

            if (gameState == GameState.GamePlay20 || gameState == GameState.GamePlay21)
            {
                spriteBatch.Draw(emotion[20], leftImgRec, leftImgColor);
                spriteBatch.Draw(emotion[21], rightImgRec, rightImgColor);
                spriteBatch.Draw(text[10], textRec, Color.White);
                spriteBatch.Draw(game[2], gameRec, Color.White);
            }


            if (gameState == GameState.GamePlay22 || gameState == GameState.GamePlay23)
            {
                spriteBatch.Draw(emotion[22], leftImgRec, leftImgColor);
                spriteBatch.Draw(emotion[23], rightImgRec, rightImgColor);
                spriteBatch.Draw(text[11], textRec, Color.White);
                spriteBatch.Draw(game[2], gameRec, Color.White);
            }

            if (gameState == GameState.GamePlay24 || gameState == GameState.GamePlay25)
            {
                spriteBatch.Draw(emotion[24], leftImgRec, leftImgColor);
                spriteBatch.Draw(emotion[25], rightImgRec, rightImgColor);
                spriteBatch.Draw(text[12], textRec, Color.White);
                spriteBatch.Draw(game[2], gameRec, Color.White);
            }

            if (gameState == GameState.GamePlay26 || gameState == GameState.GamePlay27)
            {
                spriteBatch.Draw(emotion[26], leftImgRec, leftImgColor);
                spriteBatch.Draw(emotion[27], rightImgRec, rightImgColor);
                spriteBatch.Draw(text[13], textRec, Color.White);
                spriteBatch.Draw(game[2], gameRec, Color.White);
            }

            if (gameState == GameState.GamePlay28 || gameState == GameState.GamePlay29)
            {
                spriteBatch.Draw(emotion[28], leftImgRec, leftImgColor);
                spriteBatch.Draw(emotion[29], rightImgRec, rightImgColor);
                spriteBatch.Draw(text[14], textRec, Color.White);
                spriteBatch.Draw(game[2], gameRec, Color.White);
            }

            if (gameState == GameState.GamePlay30)
            {
                spriteBatch.Draw(last, lastRec, Color.White);
            }


            // TODO: Add your drawing code here


        // kinect -----> arxi

            //call DrwaSkeleton method
            DrawSkeleton(spriteBatch, new Vector2(graphics.PreferredBackBufferWidth,
graphics.PreferredBackBufferHeight), jointTexture);

        // kinect ------>telos
```

```
            spriteBatch.End();

            base.Draw(gameTime);
        }


    // kinect ------> arxi


        //convert the depth frame data into an image
        private byte[] ConvertDepthFrame(short[] depthFrame, DepthImageStream depthStream)
        {
            int RedIndex = 0, GreenIndex = 1, BlueIndex = 2, AlphaIndex = 3;

            //a byte array to return our data in a 32bit format
            //we calculate its size based on the width and height of the depth frame image
            byte[] depthFrame32 = new byte[depthStream.FrameWidth * depthStream.FrameHeight * 4];

            //Looping through the image we take each short of information and convert it to a byte with a bit shift right
            for (int i16 = 0, i32 = 0; i16 < depthFrame.Length && i32 < depthFrame32.Length; i16++, i32 += 4)
            {
                // transform 13-bit depth information into an 8-bit intensity appropriate
                // for display (we disregard information in most significant bit)
                int player = depthFrame[i16] & DepthImageFrame.PlayerIndexBitmask;
                int realDepth = depthFrame[i16] >> DepthImageFrame.PlayerIndexBitmaskWidth;


                //apply the intensity to each of the channels (RGB)
                byte intensity = (byte)(~(realDepth >> 4));
                depthFrame32[i32 + RedIndex] = (byte)(intensity);
                depthFrame32[i32 + GreenIndex] = (byte)(intensity);
                depthFrame32[i32 + BlueIndex] = (byte)(intensity);

                //make the alpha channel opaque
                depthFrame32[i32 + AlphaIndex] = 255;
            }

            //we return our byte array containing the transformed pixel data
            return depthFrame32;
        }


        void kinect_AllFramesReady(object sender, AllFramesReadyEventArgs imageFrames)
        {
            //The DepthStream returns a 16bit image so we need to convert this to 32bit

            //depth image handling (getting the depth data and converting it to an image)
            using (DepthImageFrame depthVideoFrame = imageFrames.OpenDepthImageFrame())
            {
                  //copy the pixel data to a short array
              //  short[] pixelData = new short[depthVideoFrame.PixelDataLength];

                if (depthVideoFrame != null)
                {
                  //We had taken the pixel data and stored it in our short array
                   short[] pixelData = new short[depthVideoFrame.PixelDataLength];
                   depthVideoFrame.CopyPixelDataTo(pixelData);

                   //initilaised the texture
                   depthVideo = new Texture2D(graphics.GraphicsDevice, depthVideoFrame.Width, depthVideoFrame.Height);

                   //set the data of depthVideo texture making a call to ConvertDepthFrame.
```

```
                depthVideo.SetData(ConvertDepthFrame(pixelData, kinect.DepthStream));
            }
        }

        //collect skeleton data from the sensor
        //we open the Skeleton Frame
        using (SkeletonFrame skeletonFrame = imageFrames.OpenSkeletonFrame())
        {
            //test if skeleton frame is null (no skeletons being tracked)
            if (skeletonFrame != null)
            {
                //we check is the skeletonData is empty or if the data from the previous frame has changed
                if ((skeletonData == null) || (this.skeletonData.Length != skeletonFrame.SkeletonArrayLength))
                {
                    //we initialise our Skeleton Data array to the correct length.
                    this.skeletonData = new Skeleton[skeletonFrame.SkeletonArrayLength];
                }
                //copy the skeleton data to our array
                skeletonFrame.CopySkeletonDataTo(this.skeletonData);
            }
        }

        //check the skeletons for their tracking state
        //check for an empty variable
        if (skeletonData != null)
        {
            //loop through each skeleton in our skeletonData array
            foreach (Skeleton skel in skeletonData)
            {
                //we check the tracking state to be Tracked
                if (skel.TrackingState == SkeletonTrackingState.Tracked)
                {
                    //copy the data to our Skeleton variable
                    skeleton = skel;
                }
            }
        }
    }


    //draw the joints for our skeleton (left hand and right hand)
    //We are passing in the SpriteBatch we want to draw the Joints with, the Resolution of our program and a Texture for the joints.
    private void DrawSkeleton(SpriteBatch spriteBatch, Vector2 resolution, Texture2D img)
    {
        if (skeleton != null)
        {
            //loop through the Joints of our skeleton and process each joint
            foreach (Joint joint in skeleton.Joints)
            {
                rightHand = skeleton.Joints[JointType.HandRight];
                leftHand = skeleton.Joints[JointType.HandLeft];
                head = skeleton.Joints[JointType.Head];

                // calculate joints coordinates
                //for both axis, we are halving the position of the joint adding a 0.5f correction
                //and repositioning it based on the resolution of our program.
                Vector2 rightHandPos = new Vector2((((0.5f * rightHand.Position.X) + 0.5f) * (resolution.X)), (((-0.5f * rightHand.Position.Y) + 0.5f) * (resolution.Y)));
                Vector2 leftHandPos = new Vector2((((0.5f * leftHand.Position.X) + 0.5f) * (resolution.X)), (((-0.5f * leftHand.Position.Y) + 0.5f) * (resolution.Y)));
```

```
            }
        }
    }

    private void Gesture()
    {
        if (skeleton != null)
        {
            if (leftHand.Position.Y > head.Position.Y && rightHand.Position.Y > head.Position.Y)
            {
                gameState = gameState + 1;
            }
        }
    }


    private void Gesture1(Joint head, Joint handLeft, Joint handRight, Joint hipCenter)
    {
        if (skeleton != null)
        {
            if (handRight.Position.Y > head.Position.Y && handLeft.Position.Y < hipCenter.Position.Y)
            {
                rightImgColor = Color.LightGreen;
                gameState = gameState + 1;
            }
            else if (handLeft.Position.Y > head.Position.Y && handRight.Position.Y < hipCenter.Position.Y)
            {
                leftImgColor = Color.LightPink;
            }
        }
    }


    private void Gesture2(Joint head, Joint handLeft, Joint handRight, Joint hipCenter)
    {
        if (skeleton != null)
        {
            if (handRight.Position.Y > head.Position.Y && handLeft.Position.Y < hipCenter.Position.Y)
            {
                rightImgColor = Color.LightPink;
            }
            else if (handLeft.Position.Y > head.Position.Y && handRight.Position.Y < hipCenter.Position.Y)
            {
                leftImgColor = Color.LightGreen;
                gameState = gameState + 1;
            }
        }
    }

// kinect ------> telos


    }
}
```

3. Presentation

Gesture-based Serious Game με θέμα την αναγνώριση των συναισθημάτων από τις εκφράσεις του προσώπου για παιδιά με αυτισμό.

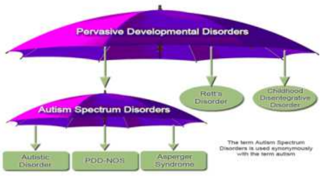
Τα άτομα με αυτισμό σκέφτονται διαφορετικά και έχουν έναν ιδιαίτερο τρόπο αντίληψης και κατανόησης των ανθρώπων και του περιβάλλοντος.

Αυτισμός

- Προέρχεται από την ελληνική λέξη «εαυτός».
  - ✓ Υποδηλώνει απομόνωση ενός ατόμου στον εαυτό του.
- Ομάδα διαταραχών που σχετίζονται με την ανάπτυξη του εγκεφάλου.
- Παρουσιάζεται πιο συχνά στα αγόρια από ότι στα κορίτσια (4:1).
- Τα ακριβή αίτια δεν είναι ακόμη γνωστά.
  - ✓ Υπάρχουν ενδείξεις ότι πιθανόν να προκύπτει από περισσότερες από μια αιτίες (γενετικοί, βιολογικοί και περιβαλλοντικοί παράγοντες).
- Εμφανίζεται συνήθως πριν τη συμπλήρωση των τριών πρώτων χρόνων της ζωής του παιδιού (DSM-IV-TR, ICD-10).

Διάχυτες Αναπτυξιακές Διαταραχές - Φάσμα του Αυτισμού

- Ο αυτισμός προσδιορίζεται ως μια σοβαρή Διάχυτη Αναπτυξιακή Διαταραχή.
  - ✓ Διάχυτη = επηρεάζει περισσότερους από έναν τομείς ανάπτυξης.
  - ✓ Αναπτυξιακή = παρουσιάζει αποκλίσεις και καθυστερήσεις στην ψυχοκινητική ανάπτυξη των παιδιών.
- Είναι ο πιο συχνός και κύριος εκπρόσωπος της ομάδας των Διαταραχών στο Φάσμα του Αυτισμού.



Επικράτηση αυτισμού - τριάδα διαταραχών

- Πρόσφατες επιδημιολογικές έρευνες παρουσιάζουν σημαντική αύξηση στα ποσοστά (2006-2008 αύξηση 23% και 2002-2008 αύξηση 78%).
  - ✓ Σε παγκόσμιο επίπεδο είναι 62/10.000 (περίπου 1/160) η εκτιμώμενη επικράτηση των Διαταραχών στο Φάσμα του Αυτισμού.
  - ✓ Στην Ελλάδα τουλάχιστον 20.000-30.000 άτομα με διαταραχές ανάπτυξης αυτιστικού τύπου.
- Ο αυτισμός επηρεάζει τρεις περιοχές της ανάπτυξης:
  - ✓ Κοινωνική αλληλεπίδραση
  - ✓ Επικοινωνία
  - ✓ Δημιουργική φαντασία
- Περιλαμβάνει ένα ευρύ φάσμα συμπτωμάτων και παρουσιάζει ποικιλομορφία στην κλινική εικόνα.

Κοινωνική αλληλεπίδραση και αυτισμός (1/2)

- Ο αυτισμός χαρακτηρίζεται από ποιοτικές αποκλίσεις στην κοινωνική αλληλεπίδραση.
  - ✓ Ποιοτικές δυσκολίες στην κοινωνική κατανόηση, συναλλαγή και συναισθηματική αμοιβαιότητα.
  - ✓ Έλλειμμα στην κοινωνική αλληλεπίδραση δημιουργεί πρόβλημα στην προσαρμογή των ατόμων στα κοινωνικά δεδομένα.
  - ✓ Προβλήματα στις κοινωνικές σχέσεις λόγω των απρόσφορων, ακατάλληλων και προβληματικών συμπεριφορών του παιδιού.
- Βασικό χαρακτηριστικό του αυτισμού σε σχέση με την κοινωνική αλληλεπίδραση:
  - ✓ Αδυναμία των ατόμων να αναγνωρίσουν και να κατανοήσουν τις δικές του συναισθηματικές και νοητικές καταστάσεις καθώς και των άλλων ατόμων γύρω τους.

Κοινωνική αλληλεπίδραση και αυτισμός (2/2)

- Η κατανόηση των συναισθημάτων απορρέει από το συνδυασμό διαφορετικών πληροφοριών:
  - ✓ Ανταλλαγή λεκτικών και μη-λεκτικών μηνυμάτων, με σκοπό τα άτομα να παρέχουν και να λαμβάνουν πληροφορίες για τον εαυτό τους και για τους άλλους.
- Χρησιμοποιούν ελάχιστες μη-λεκτικές εκφράσεις για να δηλώσουν τα δικά τους συναισθήματα.
- Οι εκφράσεις του προσώπου και οι χειρονομίες τους είναι παράξενες, μηχανικές και δεν έχουν επικοινωνιακό χαρακτήρα.
- Το πρόσωπο είναι ο «καθρέπτης» της συναισθηματικής κατάστασης.
  - ✓ Οι εκφράσεις του προσώπου αποτελούν την εξωτερική εκδήλωση της εσωτερικής συναισθηματικής κατάστασης των ατόμων.

Διδασκαλία συναισθημάτων και αυτισμός

- Ένα σημαντικό ποσοστό ατόμων με αυτισμό παρουσιάζει βελτίωση στους τομείς των χαρακτηριστικών διαταραχών μετά από εκπαίδευση.
- Οι συναισθηματικές και κοινωνικές δεξιότητες των παιδιών αυτών μπορούν να καλλιεργηθούν.
- Η εκπαίδευση μπορεί και πρέπει να ξεκινάει από πολύ μικρή ηλικία.
  - ✓ Λόγω της πλαστικότητας του εγκεφάλου που επιτρέπει τη βέλτιστη αφομοίωση των ερεθισμάτων.
- Η πρώιμη παρέμβαση είναι ιδιαίτερα σημαντική για τα παιδιά με αυτισμό.
  - ✓ Βελτιώνει τη λειτουργικότητα τους και την ποιότητα της ζωής τους και της οικογένειάς τους.
- Τα παιδιά ηλικίας 2-6 ετών μπορούν να παρακολουθήσουν πρόγραμμα πρώιμης παρέμβασης.

Serious Games και αυτισμός (1/2)

- Serious Games = ηλεκτρονικά παιχνίδια με σοβαρούς σκοπούς, όπως η διδασκαλία ή η εκπαίδευση.
- *"Serious Games: a mental contest, played with a computer in accordance with specific rules, that uses entertainment to further government or corporate training, education, health, public policy, and strategic communication objectives"* ***Michael Zyda (2005).***
- Προγράμματα με υπολογιστή έχουν ευρέος χρησιμοποιηθεί με επιτυχία για να διδάξουν την αναγνώριση των συναισθημάτων σε άτομα με αυτισμό.
  - ✓ Είναι κατάλληλα γιατί μπορούν να ενσωματώσουν πρακτικές από όλες τις παραδοσιακές προσεγγίσεις.
  - ✓ Έχουν θετική επιρροή στα παιδιά με αυτισμό και παρακινούν τα άτομα να μάθουν περισσότερα.

Serious Games και αυτισμός (2/2)

- Οι παρεμβάσεις με τη χρήση υπολογιστή απαιτούν σημαντικές δεξιότητες για την αλληλεπίδραση.
  - ✓ Δεξιότητες για το χειρισμό του ποντικιού ή/και του πληκτρολογίου.
- Προηγούμενες μελέτες που χρησιμοποίησαν τα Serious Games με τα παιδιά με αυτισμό, λάμβαναν ως δεδομένες τις απαιτούμενες ικανότητες χρήσης του ποντικιού ή/και του πληκτρολογίου.
- Αυτές οι ικανότητες είναι πέρα από τις δυνατότητες που έχουν τα πολύ μικρά παιδιά με αυτισμό.
  - ✓ Αντιμετωπίζουν σοβαρούς περιορισμούς στην ικανότητα αλληλεπίδρασης με τον υπολογιστή.
  - ✓ Πριν την ηλικία των 6 ετών, τα περισσότερα παιδιά με αυτισμό δεν έχουν αυτές τις βασικές δεξιότητες.

Serious Game - εκμάθηση συναισθημάτων - φυσική αλληλεπίδραση

- Δημιουργήσαμε ένα εκπαιδευτικό παιχνίδι:
  - ✓ Θέμα παιχνιδιού η αναγνώριση συναισθημάτων από τις εκφράσεις του προσώπου.
  - ✓ Βασίζεται στις αρχές των Serious Games.
  - ✓ Λαμβάνει υπόψη τα ειδικά χαρακτηριστικά του αυτισμού.
  - ✓ Εργαλείο πρώιμης παρέμβασης.
  - ✓ Ειδικά σχεδιασμένο να ανταποκρίνεται στις ανάγκες των παιδιών προσχολικής ηλικίας με αυτισμό.
  - ✓ Ελεγχόμενο με χειρονομίες (φυσική αλληλεπίδραση).
- Κύριος σκοπός να στηρίξουμε τα άτομα με αυτισμό, τις οικογένειες τους και τους εκπαιδευτές τους.
  - ✓ Να ενισχύσουμε τη διδακτική διαδικασία.
  - ✓ Να βοηθήσουμε την κοινωνική ένταξη των παιδιών.

Περιβάλλον παιχνιδιού(1/2)

- Είναι πολύ απλό και με το δυνατόν λιγότερες λεπτομέρειες.
  - ✓ Για να αποφευχθεί η απόσπαση της προσοχής.
- Μαύρα γράμματα και κείμενα σε λευκό φόντο.
  - ✓ Αντίθεση άσπρο-μαύρο για αύξηση και διατήρηση προσοχής καθώς και συγκέντρωση στην οθόνη.
- Ηχητικές οδηγίες – πληροφορίες.
- Γκρι ερέθισμα (εικόνες).
- Ανοιχτό πράσινο - σωστή απάντηση.
- Ανοιχτό κόκκινο - λάθος απάντηση.

Περιβάλλον παιχνιδιού (2/2)

- Δομημένο μαθησιακό περιβάλλον αποτελούμενο από τρία επίπεδα με αυξανόμενη δυσκολία.
  - ✓ Μικρότερα βήματα κάνουν τη διαδικασία ευκολότερη να εκτελεστεί.
- Τα τρία επίπεδα παρέχουν για τα συναισθήματα από τις εκφράσεις του προσώπου:
  - ✓ Αναγνώριση, Αντιστοίχηση, Παρατήρηση, Κατανόηση, Γενίκευση.
- Για τα μικρά παιδιά είναι σκόπιμο να ξεκινήσουμε με τα βασικά συναισθήματα:
  - ✓ Χαρά, Λύπη, Θυμός, Έκπληξη, Φόβος
- Τα τυπικός αναπτυσσόμενα παιδιά αναγνωρίζουν και καταλαβαίνουν τα βασικά συναισθήματα σε ηλικία 2-7 χρονών.

Ηχητικές οδηγίες - πληροφορίες

- Οι πληροφορίες που παρουσιάζονται είναι ξεκάθαρες και κατάλληλες για την ηλικία των παιδιών.
- Ηχητική απόκριση συστήματος.
  - ✓ Λάθος απάντηση:
    - «Προσπάθησε πάλι» για ενθάρρυνση.
  - ✓ Σωστή απάντηση:
    - «Μπράβο» για επιβράβευση.
- Δεν υπάρχουν άλλα ηχητικά εφέ.
  - ✓ Υπερευαισθησία σε ορισμένους ήχους (ακόμη και χαμηλής έντασης).
  - ✓ Αισθάνονται δυσφορία όταν εκτεθούν σε ορισμένους ήχους.

Εικόνες – ερέθισμα (1/2)

- Τα άτομα με αυτισμό είναι συνήθως οπτικοί τύποι.
  - ✓ Κατανοούν και επεξεργάζονται τις γραπτές λέξεις, τις φωτογραφίες και τις οπτικές πληροφορίες καλύτερα από την ομιλία.
    - Είναι καλό οι πληροφορίες να παρουσιάζονται μέσω της ισχυρότερής τους περιοχή επεξεργασίας.
  - ✓ Τα συναισθήματα παρουσιάζονται με εικόνες:
    - Χωρίς περίγραμμα.
    - Χωρίς χρώμα.
    - Με ελάχιστες λεπτομέρειες.
- Απλή και συγκεκριμένη επεξήγηση των συναισθημάτων.

Εικόνες - ερέθισμα (2/2)

- Οι εικόνες που χρησιμοποιήσαμε είναι:
  - ✓ Grayscale φωτογραφίες.
  - ✓ Αντρικά και γυναικεία πρόσωπα.
  - ✓ CAlifornia Facial Expressions (CAFE) dataset.
  - ✓ Πληρούν τα κριτήρια FACS και έχουν πιστοποιηθεί ως FACS-correct.
- Οι εικόνες παρουσιάζονται σε κάθε δοκιμή με διαφορετικά ζεύγη φωτογραφιών.
  - ✓ Στόχος είναι η επιλογή της σωστή εικόνας.

Χρωματικές επιλογές

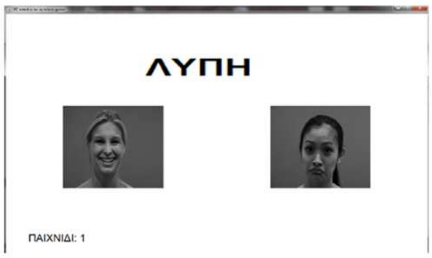
- Άσπρο, μαύρο, ανοιχτό κόκκινο, ανοιχτό πράσινο και αποχρώσεις του γκρι.
  - ✓ Μειωμένη ικανότητα διάκρισης χρωμάτων.
  - ✓ Μειωμένη χρωματική ευαισθησία.
  - ✓ Δυσκολία στην ανίχνευση διαφορών ανάμεσα στα χρώματα.
  - ✓ Μειωμένη αντίληψη των χρωμάτων.
  - ✓ Υπερευαισθησία και αυξημένη φωτοευαισθησία.
  - ✓ Προτίμηση ή αποφυγή συγκεκριμένων χρωμάτων.
    - Παράδειγμα: *Τρώνε μόνο ότι είναι κόκκινο.*

Έναρξη παιχνιδιού

- Το παιχνίδι ξεκινάει με τη σελίδα οδηγιών όπου το παιδί πληροφορείται για το τι πρόκειται να συμβεί, τι πρέπει να κάνει και πώς να το κάνει.
- Όταν το παιδί αισθανθεί έτοιμο μπορεί να επιλέξει την έναρξη του παιχνιδιού.

Πρώτο παιχνίδι – ταυτοποίηση

- Στο 1ο επίπεδο το παιδί μαθαίνει την ταυτοποίηση (labeling) των συναισθημάτων μέσω του συσχετισμού λέξης με εικόνα.



Δεύτερο παιχνίδι – αναγνώριση από τα χαρακτηριστικά του προσώπου

- Στο 2ο επίπεδο το παιδί μαθαίνει την αναγνώριση των συναισθημάτων από την περιγραφή και τη συσχέτιση τους με τα χαρακτηριστικά του προσώπου.

Τρίτο παιχνίδι – Κοινωνικές ιστορίες

- Στο 3ο επίπεδο το παιδί μαθαίνει μέσω κοινωνικών ιστοριών να αναγνωρίζει τις αιτίες των συναισθημάτων που προκύπτουν από διαφορετικές καταστάσεις.

Λήξη παιχνιδιού

- Το παιχνίδι τερματίζει με τη σελίδα τέλους όπου το παιδί ενημερώνεται για την ολοκλήρωση της διαδικασίας.

Υλοποίηση του παιχνιδιού (1/3)

- Microsoft Kinect για υπολογιστή με Windows.
  - ✓ Συσκευή εισόδου για ανίχνευση κίνησης.
    - ο Φτηνός και εύκολος τρόπος για σύλληψη κίνησης.
  - ✓ Απλή και αξιόπιστη ανίχνευση του σκελετού (skeleton tracking)
  - ✓ Ανοιχτού κώδικα SDK
- Visual Studio 2010
  - ✓ Kinect SDK 1.6
  - ✓ Γλώσσα προγραμματισμού C#
  - ✓ XNA Game Studio 4.0
    - ο Έτοιμες βιβλιοθήκες του XNA για τη δημιουργία των γραφικών

Χειρονομίες (1/2)

- Το παιχνίδι χρησιμοποιεί non-touch based NUI.
- Ελέγχεται με χειρονομίες που γίνονται με τα χέρια (hand gestures).
  - ✓ Το σύστημα αναγνωρίζει τις χειρονομίες και τις μεταφράζει σε εντολές ελέγχου.
- Ο παίκτης έχει 3 δυνατές ενέργειες σε όλα τα στάδια του παιχνιδιού:
  - ✓ Επιλογή αριστερής εικόνας
  - ✓ Επιλογή δεξιάς εικόνας
  - ✓ Επόμενη περιοχή παιχνιδιού
- Οι ενέργειες πραγματοποιούνται με αποτελεσματικές και εύκολες στην υλοποίηση χειρονομίες.

Χειρονομίες (2/2)

- Για να προχωρήσει στην επόμενη περιοχή:
  - ✓ Απαιτεί χειρονομία και με τα δύο χέρια
    - ο Μετακίνηση χεριών ψηλά, πάνω από το ύψος του κεφαλιού.
- Για την επιλογή της αριστερής εικόνας (όπως κοιτάζουμε την οθόνη):
  - ✓ Απαιτεί χειρονομία με το ένα χέρι
    - ο Μετακίνηση του αριστερού χεριού ψηλά, πάνω από το ύψος του κεφαλιού.
- Για την επιλογή της δεξιάς εικόνας (όπως κοιτάζουμε την οθόνη):
  - ✓ Απαιτεί χειρονομία με το ένα χέρι
    - ο Μετακίνηση του δεξιού χεριού ψηλά, πάνω από το ύψος του κεφαλιού.