UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR

AN INTERACTIVE SYSTEM TO ANALYZE DIETRY HABITS

-PROYECTO FIN DE CARRERA-

Carolina Camacho García

MARZO 2010

AN INTERACTIVE SYSTEM TO ANALYZE DIETRY HABITS

AUTOR: Carolina Camacho García
TUTOR: Noel O´Connor
PONENTE: José María Martínez Sánchez

Dpto. de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid

Marzo de 2010

Keywords

SenseCams, passive capture, segmentation algorithms, food pyramid, diet.

Abstract

The main objective of this PFC is the design and development of a system capable of analyze and follow the dietary habits of a real user, providing advises based on the daily diet, to be able to construct a more varied and balanced one.

For that purpose, we will use some devices: SenseCams (Microsoft). These passives wereable cameras capture an average of 1,785 images per day so they can provide a fully detailed diary of our day by day life. In this project, using algorithms and segmentation techniques, we will focus on the images where the user is eating. The system will be capable of showing the different meals in the final user screen. This user will be able to save a diary, interact with the images and even receive personalized data and dietary guides.

Palabras Clave

SenseCams, captura pasiva, algoritmos de segmentación, pirámide alimenticia, dieta.

Resumen

El objetivo de este Proyecto Fin de Carrera es el diseño y desarrollo de un sistema capaz de monitorizar y analizar los hábitos alimenticios del usuario, proporcionándole consejos basados en su dieta diaria con el fin de hacer de ésta una más sana y equilibrada.

Para ello, contaremos con unos dispositivos: SenseCams (Microsoft). Estas cámaras pasivas toman una media de 1785 imágenes por día y si se llevan colgadas del cuello, pueden proporcionarnos un diario detallado de nuestro día a día. En este trabajo, y gracias a algoritmos y técnicas de segmentación, nos centraremos en aquellas instantáneas en las que el usuario esté comiendo. El sistema será capaz de mostrar las diferentes comidas en la pantalla del usuario final, quien podrá guardar un diario de las mismas, interactuar con ellas y incluso recibir datos y guías de alimentación personalizadas

This Master Thesis has been developed in the Centre for Digital Video Processing research centre, belonging to the School of Computing and the School of Electronic Engineering.
Dublin City University. Dublin, Ireland.

Agradecimientos

Quisiera agradecer a todas las personas que han hecho posible la realización de este proyecto:

A José María Martínez, porque cuando le propuse mi proyecto en Irlanda todo fueron facilidades, y porque siempre he podido encontrarle cuando le he necesitado, durante el proyecto y durante los años de carrera.

A Noel O´Connor, que me tuvo en cuenta desde el momento en que llamé a su puerta y ha hecho posible que desarrollase este proyecto en la Dublin City University. Quisiera agradecer también a todo el equipo del Centre for Digital Video Processing por toda la ayuda prestada y muy en especial a Aiden Doherty, Aiden, for your help, your time, your patient, your ideas, your friendliness and your kindness thanks, thanks a million.

También agradecer a todo el profesorado de la Escuela Politécnica Superior, y en especial a Jesús Bescós Cano, por ser tan cercano y porque siempre nos ha llamado a todos por nuestro nombre, lo cual me parece el mayor de los piropos.

Gracias a la Universidad Autónoma de Madrid, a la O.R.I. o a quien sea que tenga que agradecer, por haber hecho posible que pasase un año de Erasmus en la Mälardalens högskola de "Vesterós", porque fue un año en el que simplemente, no podía dejar de ser feliz.

A mis compañeros y amigos de la universidad, que tantas y tantas horas hemos compartido en los últimos años, por todas las asignaturas y prácticas que aprobamos juntos, y por todos los paseos que, de vez en cuando, nos hemos dado por el "wild side of life". Mención especial merece mi "hermanamiga" Maribel, que en noches de estudio y de fiesta, con frio o calor, ha estado a mi lado haciendo posible siempre que todo siguiese adelante.

Agradecer a mi madre, Mari Beni, por su cariño, porque siempre se ha desvivido por mi bienestar y porque es bonito saber que para alguien tus alegrías, son alegrías al cuadrado; y gracias a mi hermana Elena, confidente y amiga, que aunque de pequeñas querías ser como yo, ahora soy yo la que quiere ser como tú.

También quisiera dar las gracias a mi abuelo Melchor, porque con él pisé por primera vez la universidad, con una hoja de matrícula en la mano, y a mi abuela Toni, porque nos quiere con locura, y yo a vosotros, más.

Agradezco también a Francisco Cánovas Vallés, futuro presidente, por su impagable ayuda en la realización de este proyecto y por ser fuente inagotable de ideas. Ideas tan felices como lo soy yo por estar a su lado. Gracias por los minutos, los segundos y los momentos.

No me olvido tampoco de todas esas personas con las que he compartido momentos en mi vida, que me han enriquecido y acompañado hasta el lugar donde hoy estoy.

Y por último, dedicar este trabajo a Eugenio Camacho Jiménez. Y darle las gracias. Gracias a Él por todo. Por todo, siempre.

Carolina Camacho García
MARZO 2010

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF CHARTS

1 Introduction

Having good eating habits is very important, and sometimes we are not very concerned about it. We need a varied and well- balanced diet, because there is not only one nourishment able to provide all the nutrients we need. A proper diet will increase our quality of life and will help us prevent illnesses.

However, as a result of new styles of life, and the development of new technologies in some areas related to diet, the actual society is suffering a remarkable evolution in eating habits.

Nowadays, we consume more soft drinks, soda and prepared dishes, and we also have a very low use of fruits, legumes and vegetables. That is the reason why we need every aid available to make our eating habits healthy. How about having an interactive system in our personal computer that could tell us our goods and mistakes of our diary diet?

We can use some devices: SenseCams. It is a small wearable camera. It takes approximately 3000 images per day. It has an infra-red, temperature and a light sensor as well as an accelerometer, so they can store many types of information a day.

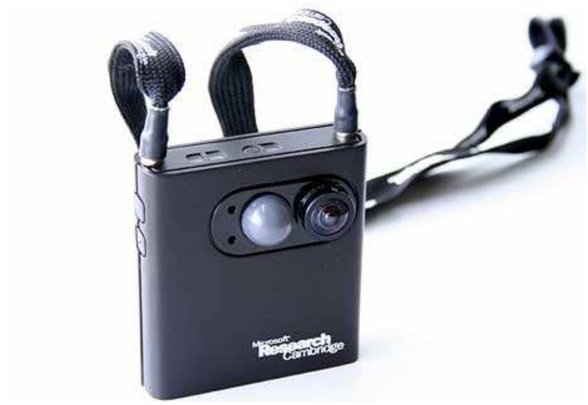


Figure 1-1: Microsoft SenseCam

This device can have many different applications: help people with dementia, make a diary of a tourist trip, or help the users have a well-balanced diet, analyzing the food that they take. We will focus in this last application in the present project.

With such a large amount of images taken, (one million of average per year) it is important to have an automated process able to segment our images in different events. This way, it would be much easier for us to find what we are looking for.

This can be done with different procedures, using MPEG-7 descriptor, with the SIFT method or with the SURF one.

Furthermore, we have to be able to recognize which of the events are related to the moments in which the user is eating. (Support Vector Machines).

We cannot forget about working with an expert in nutrition and sport. He can help us with the number of calories in each meal and the amount necessary to have a healthy nutrition, so that our Project will be as useful and efficient as possible.

1.1 Motivation and goals

The motivation of this PFC is to create a tool that can be used as a guide for the user to have a correct nutrition, using Sensecams.

In the first stage, we will study the format of the XML file in which the devices store the different data that they collect, and we will compare the color histogram to do an ideal segmentation of the events.

In the second stage, we will carry out the concept detection (SVM). This way, we will have a database with information about each of the images. We will be able to look up if one image is outdoors, indoors, a landscape…

Later, there will be a third stage in which we will develop the graphic interface, with ASP.NET. It has to be a graphic interface easy to interact with. We will use it both to take information from it and to store new data collected.

As the fourth stage, we will have to work with an expert in nutrition and sport to be able to calculate the calories taken and the ones burnt.

In the fifth stage, our project will be tested. Based on the test results, we will modify the initial program to fit the needs of real users.

Motivación y objetivos

La motivación de este PFC es crear una herramienta capaz de guiar al usuario hacia una correcta nutrición, haciendo uso de las SenseCams.

En la primera fase se estudiará el formato del fichero XML en que estos dispositivos almacenan los diferentes datos que recogen, y se compararán los histogramas de color para hacer una segmentación óptima de los eventos.

En una segunda fase, se realizará detección de conceptos (SVM). De esta forma, se dispondrá de una base de datos con información sobre cada una de las imágenes. Será posible consultar si una determinada imagen es de interiores, de exteriores, de un paisaje….y más concretamente si se está comiendo o no.

Posteriormente, habrá una tercera fase en la que se desarrollará la interfaz gráfica, por medio de ASP.NET. Debe ser una interfaz con la que el usuario pueda interactuar fácilmente. Con ella se deberá ser capaz tanto de recuperar información de la base de datos del usuario como de insertar nuevos datos aportados.

En la fase número cuatro, es el momento de trabajar con un experto en nutrición y deporte para poder calcular tanto las calorías ingeridas como las calorías gastadas por el usuario en su día a día.

En la quinta fase, probaremos el programa. Basándonos en los resultados de las pruebas, modificaremos el programa para que se ajuste a las necesidades de los usuarios reales.

1.2 Document structure

This document has the following sections:

- Chapter 1. Introduction, motivation and goals.
- Chapter 2. State of the art about SenseCams and the way they compilate data, segmentation algorithms and methods, concept detection.
- Chapter 3. Design of the system.
- Chapter 4. Development and implementation of the system.
- Chapter 5. Tests and results.

2 State of the art

2.1 The importance of learning how to eat

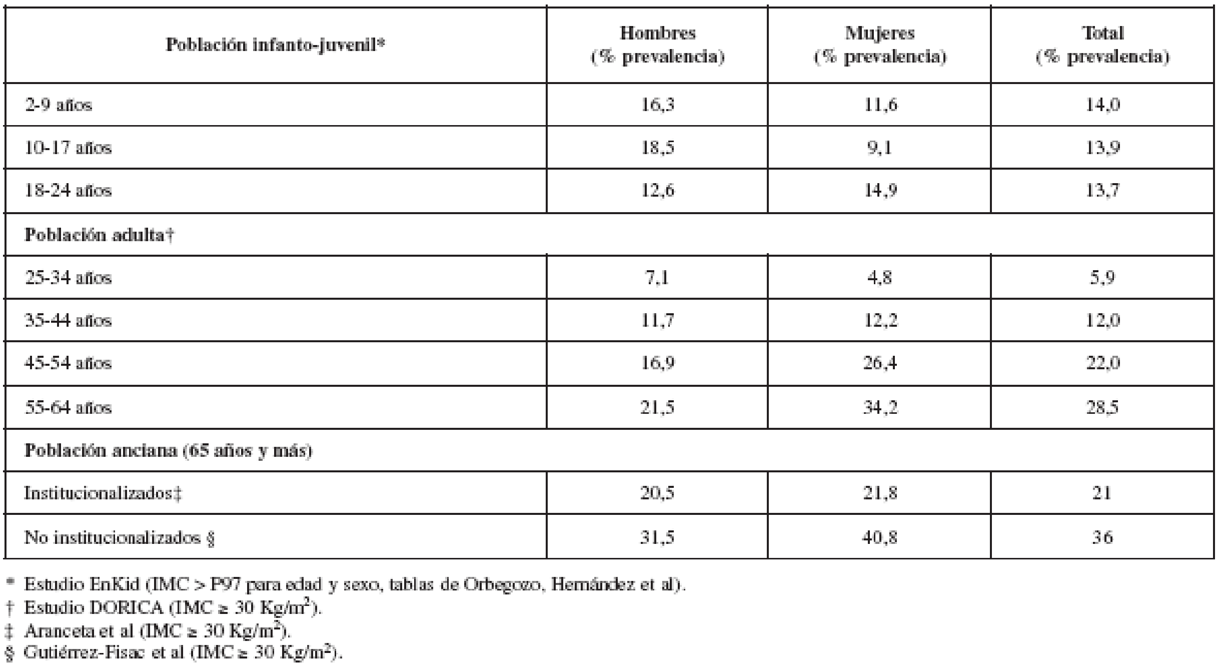
2.1.1 Introduction

Every living being needs, besides water, nutrients, vitamins and proteins, to be alive and to carry out all the basic functions that ensure the right performance of our body.

Nutrition is not the same as diet. While nutrition is the physiological process that transforms what we eat into substances that our body can assimilate, diet is the act of eating. Every living being has its own needs, a lion or a wolf, for instance, couldn´t survive eating plants. Similarly, it is not in the nature of cows, elephant or goats to take meat for their diet. Other animals, as pigs or brown bears, eat both meat and plants. This last group is called omnivorous, just like human beings are. We need a mixture of different nutrients to grow up strong and healthy.

Nevertheless, it is not that easy for us to determine what our proper needs are. If we take a look around, we can see how fast food is taking a very important place in everyone's life and how children suffer from some deceases that wouldn´t be possible if they were following a suitable diet. Therefore, **diet is something we have to learn.**

As we can see in the following table, we have to put every effort in controlling our diet and learning what our specific needs are. Obesity is a very serious problem to take into account.



Población infanto-juvenil*	**Hombres** **(% prevalencia)**	**Mujeres** **(% prevalencia)**	**Total** **(% prevalencia)**
2-9 años	16,3	11,6	14,0
10-17 años	18,5	9,1	13,9
18-24 años	12,6	14,9	13,7
Población adulta†			
25-34 años	7,1	4,8	5,9
35-44 años	11,7	12,2	12,0
45-54 años	16,9	26,4	22,0
55-64 años	21,5	34,2	28,5
Población anciana (65 años y más)			
Institucionalizados‡	20,5	21,8	21
No institucionalizados §	31,5	40,8	36

* Estudio EnKid (IMC > P97 para edad y sexo, tablas de Orbegozo, Hernández et al).
† Estudio DORICA (IMC ≥ 30 Kg/m^2).
‡ Aranceta et al (IMC ≥ 30 Kg/m^2).
§ Gutiérrez-Fisac et al (IMC ≥ 30 Kg/m^2).

Figure 2-1: Obesity in Spain.

2.1.2 Diet control

Looking good and feeling healthy can be considered an obsession in present times. We are used to read and hear tricks everywhere to lose weight and to look thinner. But reality is other thing, and these tricks are not always effective. The fact is that an equilibrate diet and appropriate exercise is the only way to be in a proper shape.

More than once, we have seen people that have diets stuck on their fridges obsessed about light products, willing to write down everything they eat to control their weight. These proposals last four or five days tops, and it usually takes so much effort that they usually give up. So how about making some automatic way of controlling our diet without taking too much time or effort from the person who wants to eat healthy? This will be an important step in the traineeship of eating. And this learning has to start since our very first years. Usually, a food pyramid is used to show people good eating habits.

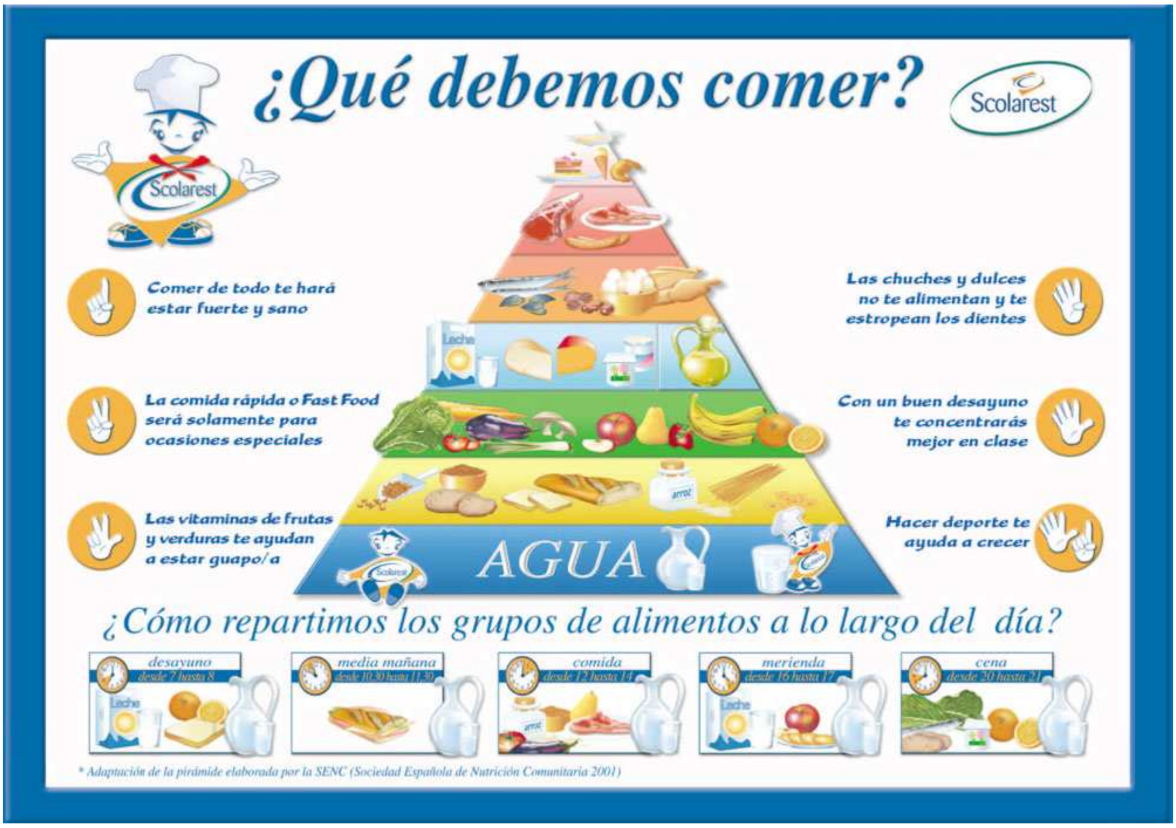

¿Qué debemos comer?
Scolarest
Scolarest
Comer de todo te hará estar fuerte y sano
La comida rápida o Fast Food será solamente para ocasiones especiales
Las vitaminas de frutas y verduras te ayudan a estar guapo/a
Las chuches y dulces no te alimentan y te estropean los dientes
Con un buen desayuno te concentrarás mejor en clase
Hacer deporte te ayuda a crecer
Leche
arroz
AGUA
¿Cómo repartimos los grupos de alimentos a lo largo del día?
desayuno
desde 7 hasta 8
media mañana
desde 10.30 hasta 11.30
comida
desde 12 hasta 14
merienda
desde 16 hasta 17
cena
desde 20 hasta 21
* Adaptación de la pirámide elaborada por la SENC (Sociedad Española de Nutrición Comunitaria 2001)

Figure 2-2: Teach children how to eat is important for their future.

Besides, we don´t have to forget about exercise as an essential part of our development. The recommended thing is to choose some sport or practice that entertains us at the same time that we exercise our body.

In the next pages, we are going to present a way to help people with their diet problems.

2.2 SenseCams and data compilation

2.2.1 Introduction

A SenseCam is a wearable digital camera. It can be hold around the neck, clipped to the belt or directly attached to different clothes. However, it is recommended to wear it around the neck because it increase stability, the right-left movements are minimized, it´s relatively comfortable for the wearer and it stays close enough to his eye line to ensure a good captation of everything that passes through his point of view.

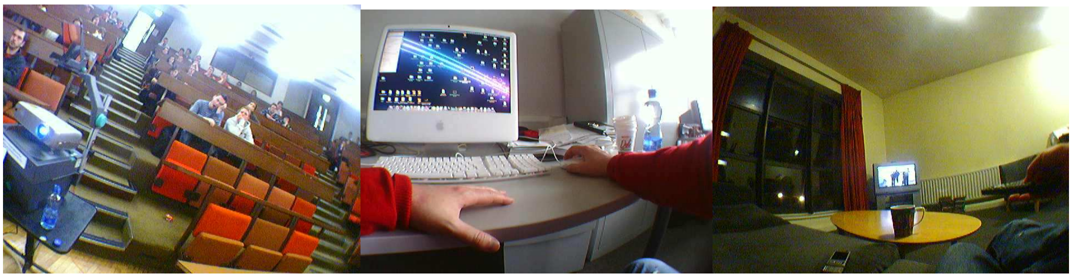


Figure 2-3: Example of SenseCam images

Unlike traditional digital cameras, it doesn´t have a display to find and look the taken photos and it is designed to take pictures in a passive way, without user interaction. It is provided with a wide-angle lens (fish-eye). This way we can be sure that most of the details of the wearer view will be recorded by the camera.

To take the photographs, the user can program a timer making the camera take a picture every, for example, 30 seconds. Alternatively, a certain event can trigger a photograph to be taken. For example, a considerable change in the light level, in the ambient temperature or a sudden movement would be noticed for one of the SenseCam sensors (light-intensity, light-color sensors, passive infrared (body heat) detector, temperature sensor or multiple-axis accelerometer) and a picture will be taken. Besides, SenseCams have a button to manually take a photo whenever the wearer thinks it´s appropriate. This way there will not be a detail that could escape from being recorded. [1]

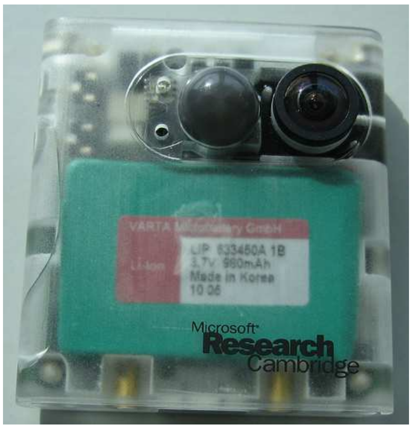


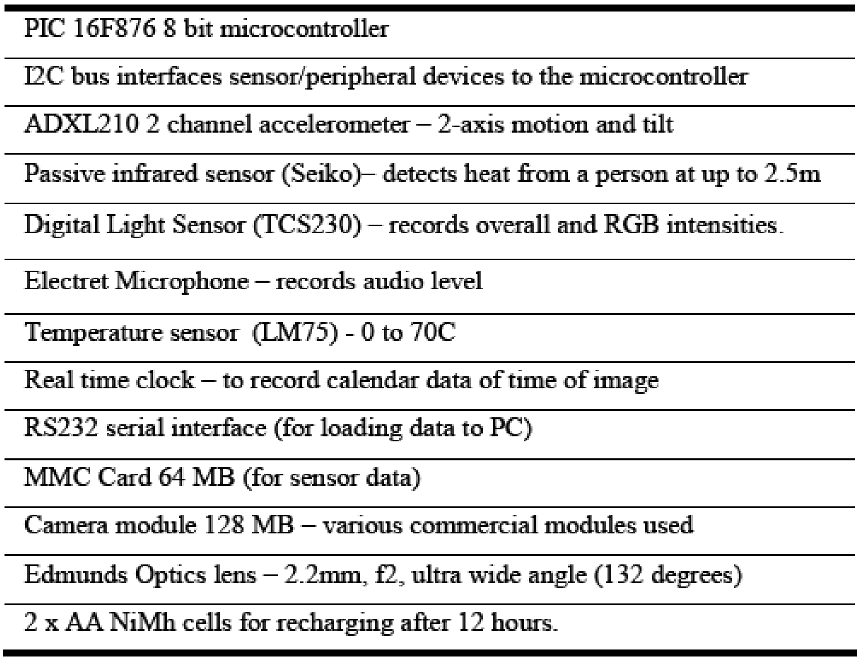
Figure 2-4: Microsoft SenseCam with transparent plastic case.

2.2.2 Data compilation

To make a captation as close to reality as possible, the SenseCams have a digital “fish-eye lens” camera and several sensors that will record a lot of different data about the wearer situation. The sensors are used to trigger pictures when there is a change in the environment and the sensor data itself will facilitate the indexal and retrieval.

The union of all these data will help to create a better understanding of a situation in every detail. All these captured data may be related altogether via time-correlation.

Table 2-1 shows the hardware present in a SenseCam.



PIC 16F876 8 bit microcontroller
I2C bus interfaces sensor/peripheral devices to the microcontroller
ADXL210 2 channel accelerometer – 2-axis motion and tilt
Passive infrared sensor (Seiko)– detects heat from a person at up to 2.5m
Digital Light Sensor (TCS230) – records overall and RGB intensities.
Electret Microphone – records audio level
Temperature sensor (LM75) - 0 to 70C
Real time clock – to record calendar data of time of image
RS232 serial interface (for loading data to PC)
MMC Card 64 MB (for sensor data)
Camera module 128 MB – various commercial modules used
Edmunds Optics lens – 2.2mm, f2, ultra wide angle (132 degrees)
2 x AA NiMh cells for recharging after 12 hours.

Table 2-1: SenseCam hardware specification [2]

2.2.2.1 SenseCam camera

The resolutions of the pictures that are taken are VGA (640x480 pixels). This is not a very high resolution, and the captured images all suffer from spherical distortion [2]. However the purpose of the SenseCam is not to create wonderful and rich media but to aid a user with some memory loss or as a simple reminder of how a certain day went by. These pictures are stored as compressed .jpg files on a flash memory placed inside the device. Due to the low resolution of the images, over 30,000 images can be fitted en a 1 Gb card. Furthermore, in the flash memory we also have place for a log file, where we can find other sensor data and their correspondent timestamps. [1]

2.2.2.2 Accelerometer

An accelerometer measures the change in velocity over time, relative to freefall. These devices are able to detect the magnitude and direction of the acceleration.
The accelerometer present in the SenseCam, can measure accelerations with a full-scale range of ±10g. It can measure both dynamic and static acceleration. To measure it, the device has outputs of analog voltage or digital signals whose ratio of pulse width to period are proportional to acceleration.

2.2.2.3 Passive infrared sensor

It´s called "passive" because it does not emit any energy. All objects emit an infrared radiation that is invisible to the human eye but can be detected by electronic devices designed for such a purpose. It is activated with the changes in the thermal energy emitted by a human "intruder" in comparison with the temperature of the object in front of which this human is passing. (A wall, for example)[5]

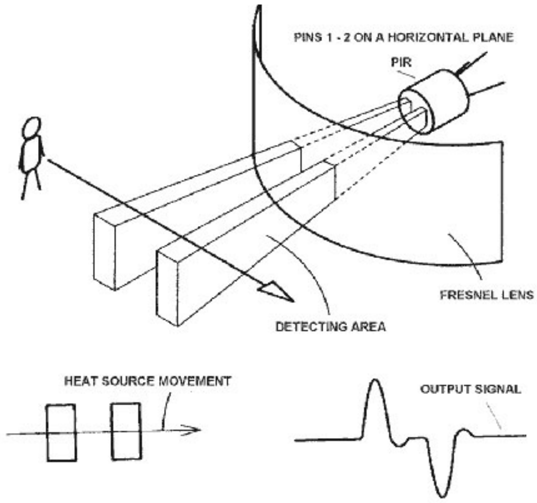


Figure 2-5: How passive infrared sensor works [7]

2.2.2.4 Digital light sensor

It has an integrated red, green and blue (RGB) filters that can derive the color chromaticity and luminance (intensity) of ambient light.

2.2.2.5 Electret microphone

The mayor advantage of these microphones (microphones with thin electrets polymer foil suspended over a perforated backplate) is that they operate without needing an external power supply and they are available at very low cost. [6]

2.2.2.6 Temperature sensor

To measure ambient temperature and notice if the wearer has moved, for example, from outdoors to indoors or viceversa. It doesn't require external components and has a shutdown mode to minimize power consumption.

2.2.2.7 Edmunds optics lens

The camera has an ultra wide-angle lens incorporated. Although it will introduce some spherical distortion to the image, the pictures taken will be closer to the real user view than a picture taken from a camera with a normal lens.

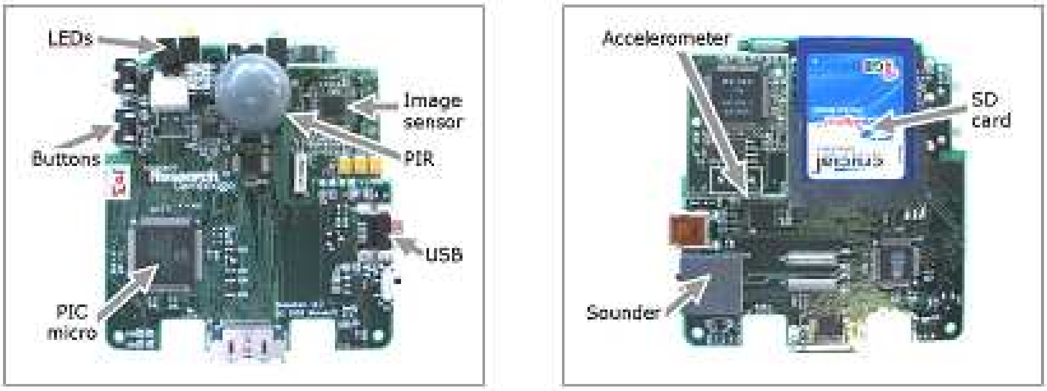
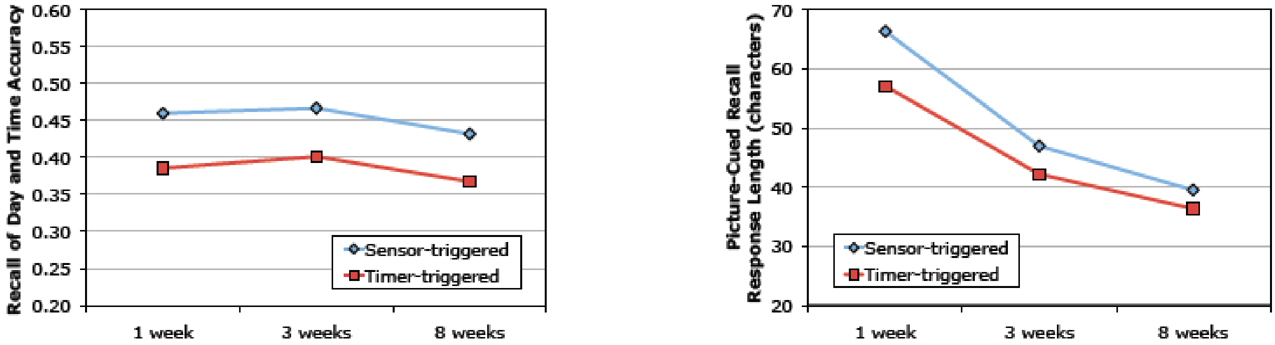


Figure 2-6: Front and back of the SenseCam PCB. [1]

Different experiments have demonstrated the importance of sensor-data as a trigger for the SenseCam. In an experiment carried out by Jason R. Finley, William F. Brewer, and Aaron S. Benjamin from the University of Illinois [3], some participants wore a SenseCam for five consecutive days. The cameras were programmed to take pictures either at fixed intervals or triggered by their sensors. On two of the five nights, the participants reviewed some of the pictures taken at the end of the day. These participants took memory tests at intervals of 1, 3 and 8 weeks. The results showed that the participants remembered clearer the days that they had previously reviewed and the images taken in sensor mode than the images taken in time mode.



Recall of day and time was more accurate for sensor-triggered than timer-triggered images.

Picture-cued recall responses were longer for sensor-triggered than timer-triggered images.

Figure 2-7: Sensor-triggered images vs. timer-triggered images.

2.3 Capture, upload and processing of the images

2.3.1 Introduction

As we pointed out before, at the end of the day, the wearer has approximately 1750 pictures in his SenseCam memory card. Thinking about organizing all this information and being able to search for a particular image doesn´t seem a very pleasant work to do manually. For example, looking through the pictures taken on a single day will take approximately 10 minutes expending just 0.3 seconds per photo.

Of course, SenseCam will not be worthy if we have to spend one hour a day just revising the images one by one and manually dividing all the different events. That is why we need an automated process to help us organize all the visual material that we will achieve at the end of each period. Here is where content-based image analysis techniques will take a very important role, helping us to distinguish the several events that took part in the development of our day.

2.3.2 Capture and upload

A SenseCam user normal day starts, of course, putting on the SenseCam. He must wear it all day, or just during some remarkable moments, from which he wants to save a graphic testimony. In our case, as we will focus on dietry habits, it would be enough to wear the SenseCam during the different meals taken in a day. However, it would be better to wear it all the time to avoid forgetting meals. Once the day is over, the camera will be connected to a PC and the photos will be uploaded. Then, automatic content-based analysis commences, making possible to organize all the different events without meaning any effort or waste of time.

2.3.3 Processing

As we explained, once the captured photos have been uploaded to a PC, different content-based image analysis techniques are applied to these photos. We will organized all pictures in a way that will be easy for the user to access them, and easier for our particular software to distinguish those photos where the user is taken food.
The processing will start with event segmentation:

2.3.3.1 Event Segmentation

The first challenge is to divide all the images taken into different groups or events and determine a boundary that means the transition between different events, for example, having lunch, watching TV, going to the supermarket, etc. To be able to do so, we will have to use context-based sensor analysis techniques in conjunction with content-based image analysis. [8]

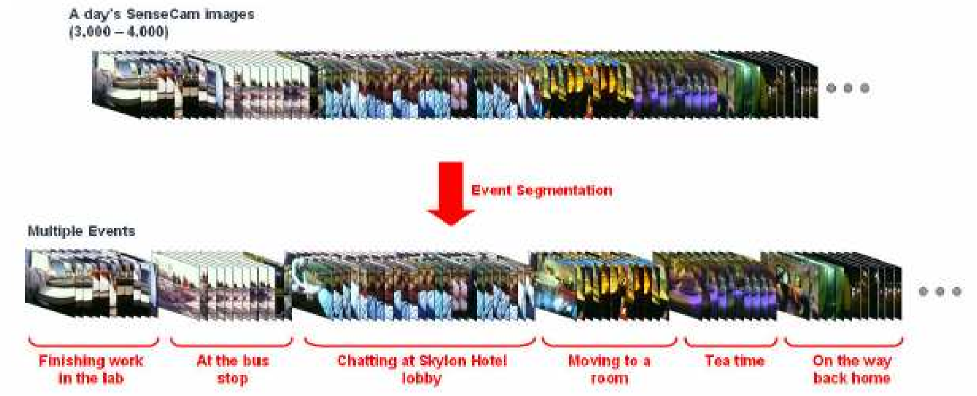


Figure 2-8: Segmenting images into events.

Talking about SenseCams, if we compare adjacent images, we can find two very different images inside the same event. This is because the images are taken in a low frequency, compared, for example with the rate of images taken for a video camera, and a photo could be taken when the user turn himself for a moment, while sitting in front of the computer. This is the most common reason to trigger false events: slightly change of position of the user.

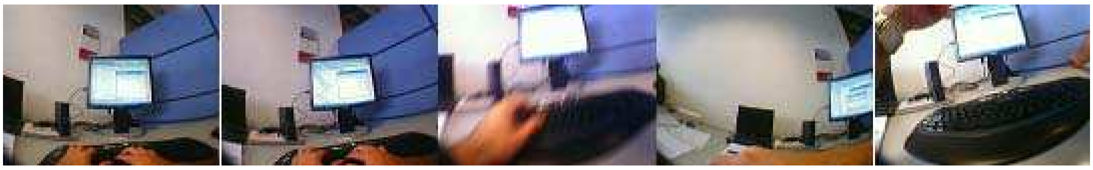


Figure 2-9: Examples of false positive boundary [11]

To segment a group of images into events using content-based image analysis, an adaptation of Heart´s Text Tiling approach is used. [9] With this technique, we take a reference image, and then compare the block of images previous to it with the block of images that come afterwards. Each block is represented with the average value of the low level MPEG-7 visual features of all images present in that block. This way we can solve the problem of “intruder” images inside a certain event.

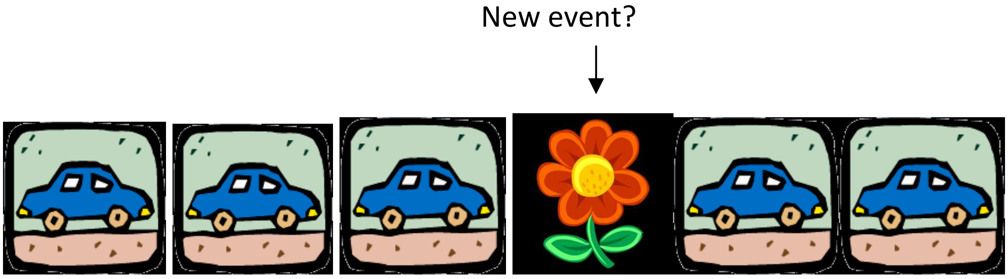


Figure 2-10: The only flower image cannot be considered an event

The **MPEG-7** visual features are color structure, color layout, scalable color and edge histogram. They are calculated making use of the aceToolbox [18].

Color Layout Descriptor (CLD): it is resolution-invariant and it is designed to capture the spatial distribution of color in an image or an arbitrary-shaped region. To extract the color layout descriptor, firstly the image is partitioned into 8x8=64 blocks. Each block is representated by its average color. This result in three 8x8 arrays, one for each color component (Y: luminance. Cb: Blue-difference chrominance components. Cr: red-difference chrominance components). Then, the DCT transform is applied. We now have 3 matrices of coefficients. The resulting coefficients are zig-zag-scanned and the CLD descriptor is formed by only 6 coefficients from the Y-DCT-matrix and 3 coefficients from each DCT matrix of the two chrominance components. The Descriptor is saved as an array of 12 values. Finally, the remaining coefficients are nonlinearly quantized. [10], [11]

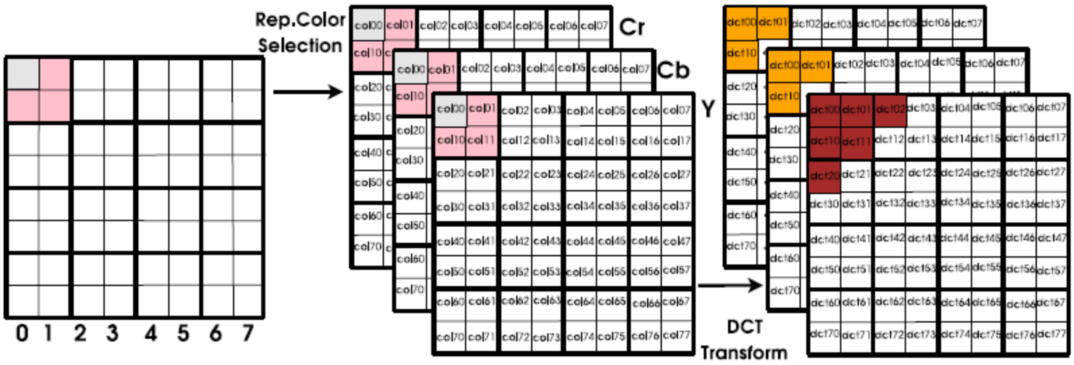


Figure 2-11: The CLD extraction process – Steps [10]

Scalable Color Descriptor (SCD): this is a color histogram extracted in HSV (Hue, saturation and value) color space computated with 256 bins. To reduce the large size of this representation the Haar-transform encoding scheme is used. The SCD measures the color distribution over an entire image. It´s often used to look for similarities in multimedia databases and for browsing in large databases. [12]

Edge Histogram Descriptor (EHD): It is scale invariant and represents the spatial distribution of five types of edges: vertical, horizontal, 45°, 135° and non-directional. The input image is divided in 16 (4x4) non-overlapping blocks and a 5-bin edge histogram of each subimage can be obtained. Each bin value is normalized by the total number of image-blocks and finally, the normalized bin values are nonlinearly quantized. This descriptor provides a better representation of event boundaries as it was less sensitive to Changes in light than scalable color descriptor. [11]

Color structure (CSD) : It expresses local color structure in an image by use of a structuring element. The CSD is calculated by taking the colors present in each of the pixel that form the 8x8 pixel structure. Each color is identified (C0-C7) and the correspondent CSD bins assign to each color suffers an increment for each positive id. This method is equivalent to subsampling the image by the power of 2 and then using the structuring element on it.[13]

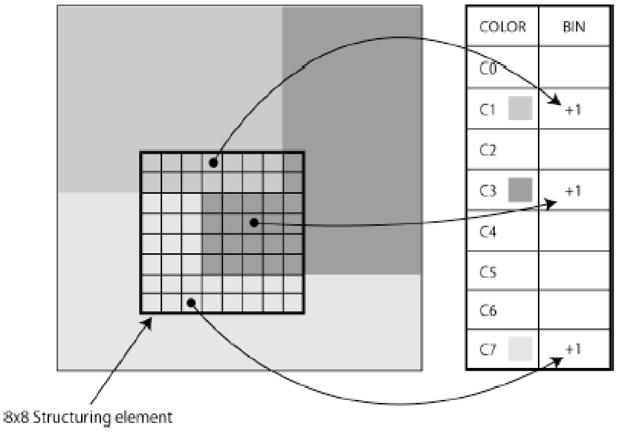


Figure 2-12: CSD structuring element [13]

We can also segment photos using the information that the different sensors provide us. This can be called context-based analysis. The values of sensors can indicate the probability that the wearer may have changed activities. For example, a change in temperature or in the light sensors can indicate that the user has gone from indoors to outdoors or specific values in the infrared sensors can indicate that a person has crossed in front of the wearer. The optimal way to separate the events of a day will be by fusing the output of the content based image analysis with the output corresponding to the context-based sensor analysis.

Therefore, to proceed with the segmentation of events, we will have to focus on the information retrieved with our SenseCam, the results obtained after some image processing and the best way to calculate the similarity between images: the metric that will provide optimal results and the calculation of appropriate threshold.

To segment all the images taken in a day in different events, we have to follow some steps:

1. Revising the time-stamps, if found that the camera has been disconnected more than two hours, it will be automatically a boundary of different events. It is usually registered as a day change, as the wearer would have disconnected the camera to go to sleep.
2. **Compare adjacent images against each other** (**or blocks of images) to see how dissimilar they are.**
   The TextTiling approach [9] was found to perform better on average than non-TextTiling for the MPEG-7, passive infrared and temperature data sources. However, not the same size of blocks of images were optimal for every technique, resulting in an average value of 5 images grouped together for the MPEG-7 source and 8 images for the temperature and passive infrared one. This is because temperature and passive infrared values change slower over time (the temperature showed by the sensor would remain almost constant if the wearer do not change his location). However, if we have a source of information which values change quickly over time, it would be better to compare the values of adjacent images. This could be the case of data from accelerometers and light sensors.

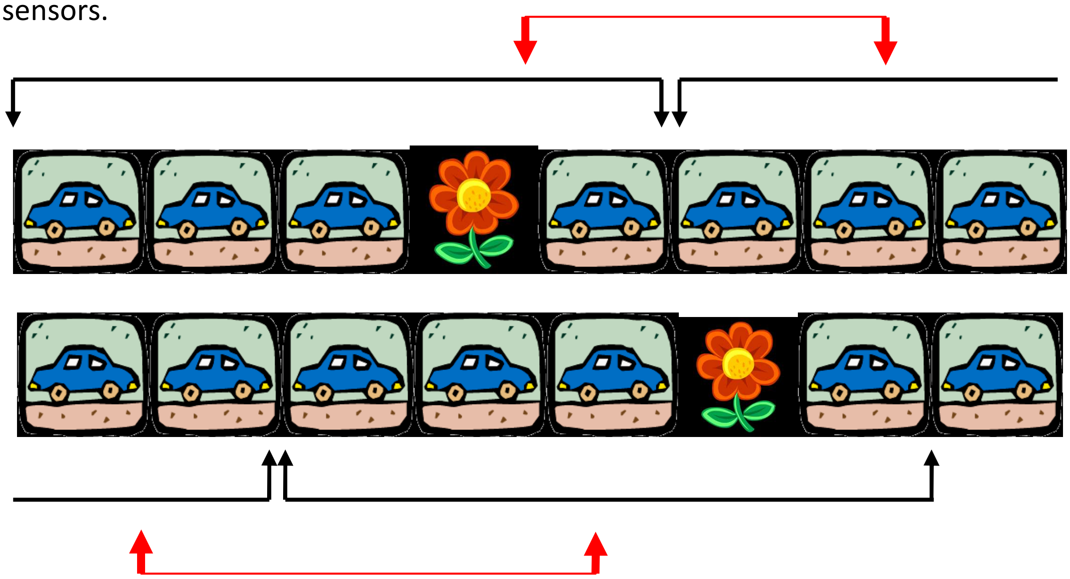


Figure 2-13: Comparison of blocks of images. All images belong to the same event.

3. **Determine a threshold value.** Images with dissimilarity values higher than the threshold value will be with probability a boundary between different events. Some of the techniques to select boundaries are non-parametric (kapur) and parametric (Mean).[14]

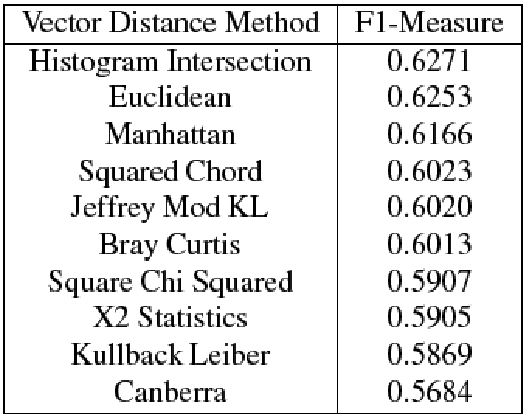
4. **Remove two event boundaries too close in time to each other**.
   Sometimes, if we are moving quickly, for example riding a bicycle, our SenseCam will take pictures that will be very different to each other although they belong to the same event. Only one image has to be considered as a boundary. But what is the optimal time under which considered boundaries have to be ignored? The people from DCU have determined with experimentation that the gap of 3 minutes is the best. [15]

The **MPEG-7** visual features are color structure, color layout, scalable color and edge histogram. Once the image is processed as explained before and all the MPEG-7 visual features are extracted, we have to proceed to compare images characteristics, and for that we will choose the best metric for our purpose. In Figure 2-14, we can see how the visual features of each picture are represented, using Extensible Markup Language to facilitate its manipulation (**Appendix A**). In the figure, the events are already separated and marked.

```
<ImageInfoCollection>
<Image
blueRegionAverage="14.217396_12.927767_4.7789097_13.228907_10.220811_11.8553
81_1.5571094_2.4198046_2.2613835" bookmark="DCU Event 0"
colourAverage="45.71277_59.83522_73.47109"
colourHistogram="121_106_43_74_301_620_718_1411_2637_4554_44815_18706_1150
_232_103_88_34_14_1315_109_112_182_1033_820_15844_3279_5040_11195_33203_
3664_1260_378_77_87_115_96_63_0_0_0_0_0_183_532_1444_6412_28961_390_0_0_
0_0_0_0_38450_88_696_546_139_61_48_56_97_893_35750_203_61_9_4_6_22_22_38
528_0_0" distance="0"
edgeHistogram="0_1_0_0_0_0_0_19_0_0_0_0_0_0_4_2_1_3_0_1_7_2_37_70_1_8_0_1
_1_4_70_54_0_7_1_0_6_0_40_94_1_5_0_1_2_3_21_116_0_10_0_0_5_0_23_11_0_0_0
_0_2_2_6_64_0_19_0_1_1_0_0_22_0_1_0_0_0_0_0_2_0_26_1_1_8_1_24_128_0_1_1_
0_1_1_0_32_0_0_0_0_0_0_1_0_0_2_0_0_0_0_9_4_0_0_0_0_0_0_7_0_0_0_0_0_0_0_2
9_0_0_0_0_0_0_0_0_0_0_0_0_0_0_13_0" filename="00352614.JPG"
greenRegionAverage="11.300309_9.423034_4.6796713_11.157379_8.149681_9.996442
_1.3440951_1.8769531_1.9110873"
redRegionAverage="9.255179_7.45305_2.8036199_8.652275_6.0556545_7.7606444_0.
80875_1.3782878_1.545306" timestamp="2008-10-06T10:44:04"/>
</ImageInfoCollection>
```

Figure 2-14: Visual features of an image, represented in XML

There are many different metrics that we can choose to compare all the images and to calculate how dissimilar they are. As we can read in [15] several metrics have been tested to find out which one has the best results:



Vector Distance Method	F1-Measure
Histogram Intersection	0.6271
Euclidean	0.6253
Manhattan	0.6166
Squared Chord	0.6023
Jeffrey Mod KL	0.6020
Bray Curtis	0.6013
Square Chi Squared	0.5907
X2 Statistics	0.5905
Kullback Leiber	0.5869
Canberra	0.5684

Table 2-2: Vector distance methods results

The results are reported in terms of the F1-measure. This is a measure of the accuracy of a test. It is considered the precision (p) and the recall (r).

$$p = \frac{correct\ results}{all\ results} \qquad r = \frac{correct\ results}{results\ that\ should\ have\ been\ returned}$$ [1]

$$F = \frac{2 \cdot (precision \cdot recall)}{precision + recall}$$

The score reaches its best value at F=1 and worst at F=0.

As we can see in Table 2-2, the best metric method based on MPEG-7 is Histogram Intersection, followed by Euclidean and Manhattan distance.

The **histogram intersection distance** compares only the elements which exist in the query.

$$H(p,q) = \frac{\sum_{j-1}^{n} min(p_j - q_j)}{\sum_{j-1}^{n} q_j}$$

The **Euclidean distance** measures the shortest distance between two points.

$$\sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}.$$

[1] This refers to the group of results that should have been returned due to their relevance.

The **Manhattan distance** is the distance between two points if a grid-like path is followed: [16]

$$d = \sum_{i=1}^{n} |p_i - q_i|$$

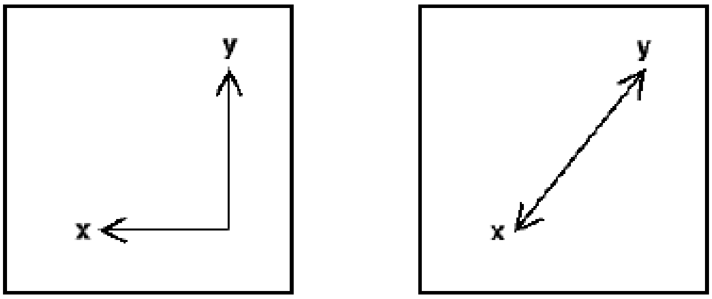
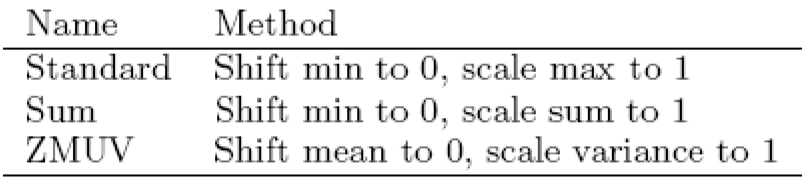


Figure 2-15: Manhattan and Euclidean distance between two points.

This last distance (Manhattan) is the one used in the study showed in Appendix A. Sensor readings are scalar values and thus the difference between sensor data of adjacent images is calculated using standard subtraction.

To calculate the overall dissimilarity score for each image, it would be necessary to fuse all the sources of information that we have: MPEG-7 descriptors, accelerometer, ambient temperature, light level and passive infrared. To do so, the values have to be *normalized.* This is because data from different sources can probably not be expressed in the same way (in different scales or range), although all data are real numbers. In the following table, some normalization algorithms are presented.

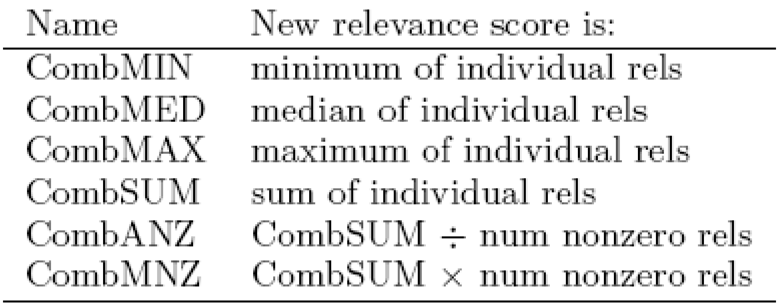


Name	Method
Standard	Shift min to 0, scale max to 1
Sum	Shift min to 0, scale sum to 1
ZMUV	Shift mean to 0, scale variance to 1

Table 2-3: Normalization algorithms

Standard and Sum are shift and scale invariants algorithms. The standard one is sensitive to the maximum and minimum score given for each data source and highly sensitive to outliers while the Sum is sensitive only to the min score and more robust. In practice the sum algorithm is fairly outlier insensitive. [17]
The ZMUV algorithm is shift and scale invariant and outlier insensitive. It´s very robust.

After normalizing the different scores, we have to *combine* all the data collected.



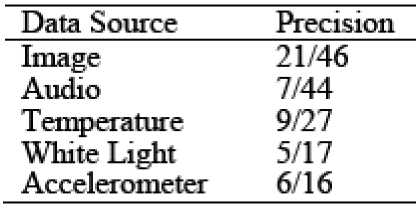
Name	New relevance score is:
CombMIN	minimum of individual rels
CombMED	median of individual rels
CombMAX	maximum of individual rels
CombSUM	sum of individual rels
CombANZ	CombSUM ÷ num nonzero rels
CombMNZ	CombSUM × num nonzero rels

Table 2-4: fusion algorithms

It is empirically determined that **Min-Max** normalization and **CombMIN** works best.
If we normalize and combine all data retrieved with our SenseCam, we will have the best approach and optimal results with the minimum amount of errors. However, the process will be slower and more difficult. If we fuse data that comes only from sensors sources the process would be much quicker and it would be worse than the fusion of MPEG-7 and sensor data in only a 3,3%. [15]

There are more ways and procedures of segmenting the data collected into events. For example, we can wear another device to record audio next to our SenseCam. However, this will not be completely useful, due to privacy concerns.

We can read some studies about how efficient will be the segmentation of events using each source of information **separately**. [19]



Data Source	Precision
Image	21/46
Audio	7/44
Temperature	9/27
White Light	5/17
Accelerometer	6/16

Table 2-5: Precision of segmentation with different data sources.

The temperature sensor present in the SenseCam can be used to detect changes in location. However, it is very sensitive and the values can change even though the wearer is staying in the same environment. This is why the variance of the value has to be measured over a predetermined window size. If the degree of variance is higher than a certain value, it is probable that the user has change the environment, maybe going outdoors or moving between rooms. Some of the wrong boundaries were due to the time for calibration that the sensor needs once it is activated and due to temperature changes in the same environment: a cloud covering the Sun if outdoors, the air conditioning turned on/off if indoors.

The light sensor simply measures the intensity of white light present. So for example, if the wearer is taking an elevator, the white light detected would probably decrease, reaching its normal value again when the elevator is left behind. These changes in the intensity of light will mark a boundary between events.

The most common falsely detected boundaries by light sensor were due to the decrease of light the sensor suffered when the wearer placed some object too close to it and the different light received from the Sun.

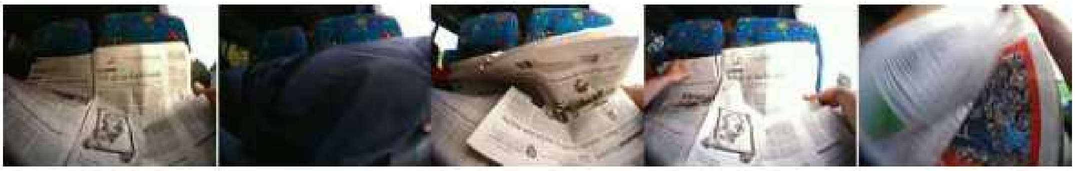


Figure 2-16: False boundary (light sensor) due to wearer movement.

Motion of the wearer is calculated by an accelerometer incorporated in the SenseCam. The data captured has 3 axes: X, Y, Z and after computing the derivative value for each axis, we use $\sqrt{X^2 + Y^2 + Z^2}$ to combine the different axis.
The boundaries falsely determined can be due to some walking around at an airport or due to motion while looking for something in a drawer.

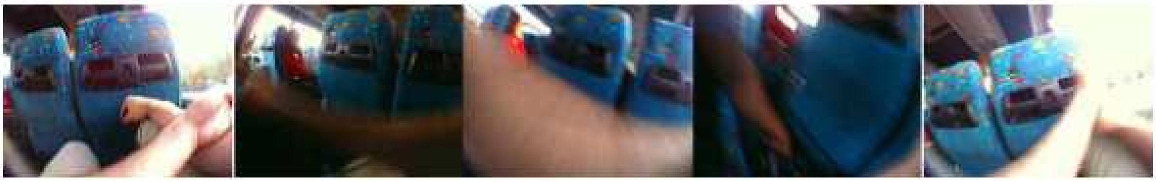
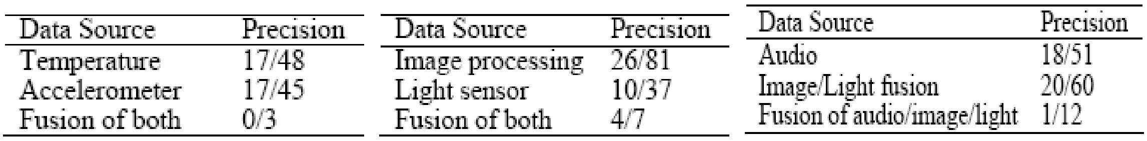


Figure 2-17: False boundary (accelerometer) due to wearer movement.

Finally, we can notice that there are three types of activity boundaries: change of activities within the same location, wearer involve in social interaction and the movement to a different location. For the first one, a fusion of image processing and light sensor will provide the most accurate results. The segmentation done with accelerometer values gives its best results for activities which implicate moving to different locations and finally, the audio sensor alone or the fusion of the image and light processing sources, provides better detection of the boundaries that implicate the wearer socializing with colleagues.[19]



Data Source	Precision
Temperature	17/48
Accelerometer	17/45
Fusion of both	0/3

Data Source	Precision
Image processing	26/81
Light sensor	10/37
Fusion of both	4/7

Data Source	Precision
Audio	18/51
Image/Light fusion	20/60
Fusion of audio/image/light	1/12

Table 2-6: Identified change location, within location and social segmentation.

2.3.3.2 Landmark photo selection

The second step in the processing of the images taken by the SenseCam involves the selection of a landmark picture for each event. An event has on average one hundred photos, so it would be helpful if we can choose just one photo that could represent the content of each event. To proceed, we have four options:

- Select the image which is closer to all other images that compose the event: this will have a computational cost of n*n. (n=number of images of an event.)
- Select the image which is closer to all other images that compose the event but that also is most distinct from all other images present in the rest of the events. This approach is computationally the most expensive: n*(n + m) (m=number of images in a day.)
- Select the image closest to the **average value** of the visual features of its own event but also as different as possible from the average values of the pictures of the rest of events. The computational cost will be n*e. (e= number of events in a day)
- Select the middle image from each event.

The most intelligent approach to use is the forth one, as it´s easier to compute and because in practice, the results doesn´t seem to be very different to the others approaches.

However, to select a representative image, we will have to consider some aspects. It´s important to choose a photo that is well-focused and that has a level of brightness that is appropriate. If it is too dark or too bright the chosen image will not be a good representative. Because of the nature of a SenseCam, a wearable and mobile device, many pictures can be taken while the wearer is moving around, resulting in a picture more fuzzy and blurry than expected. Besides, the camera can eventually move to a bad angle receiving the light directly to its lens and taken pictures too bright to be considered representative. Thus, it´s obvious that quality may play an important role in the selection of an appropriate keyframe.

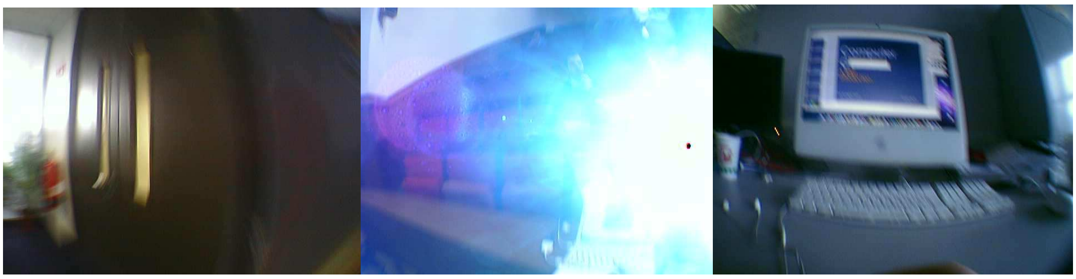


Figure 2-18: Examples of poor quality SenseCam images.

The quality measures can be automatically extracted from each image within the collection. There are five low-level image features that can measure the quality of a given image:

- **Contrast Measure**: this is the measure of the difference in brightness between the different sections in an image. Pictures taken in a place with direct sunlight will have higher contrast than pictures taken in the fog.

- **Color Variance**: here we calculate the perception of the color richness of the dominant colors in an image.

- **Global Sharpness**: it measures how sharply focused is an image. For this, we measure the sharpness based on sections of an image that are in focus.

- **Noise measure:** it is the random variation of brightness or color information in images. The noise is less pronounced in bright tones, while it is more appreciable in dark tones. It is composed for two elements: fluctuations in color and luminance.

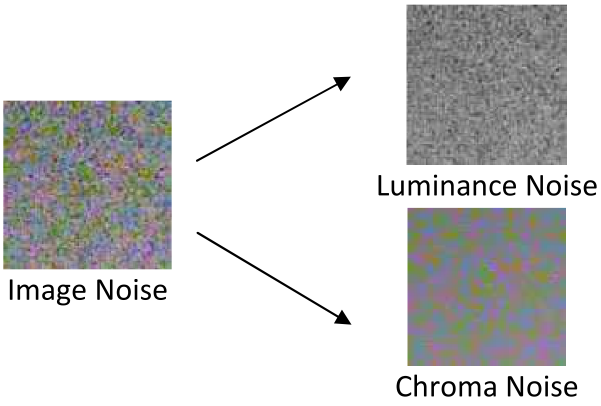


Figure 2-19: Noise composition [21]

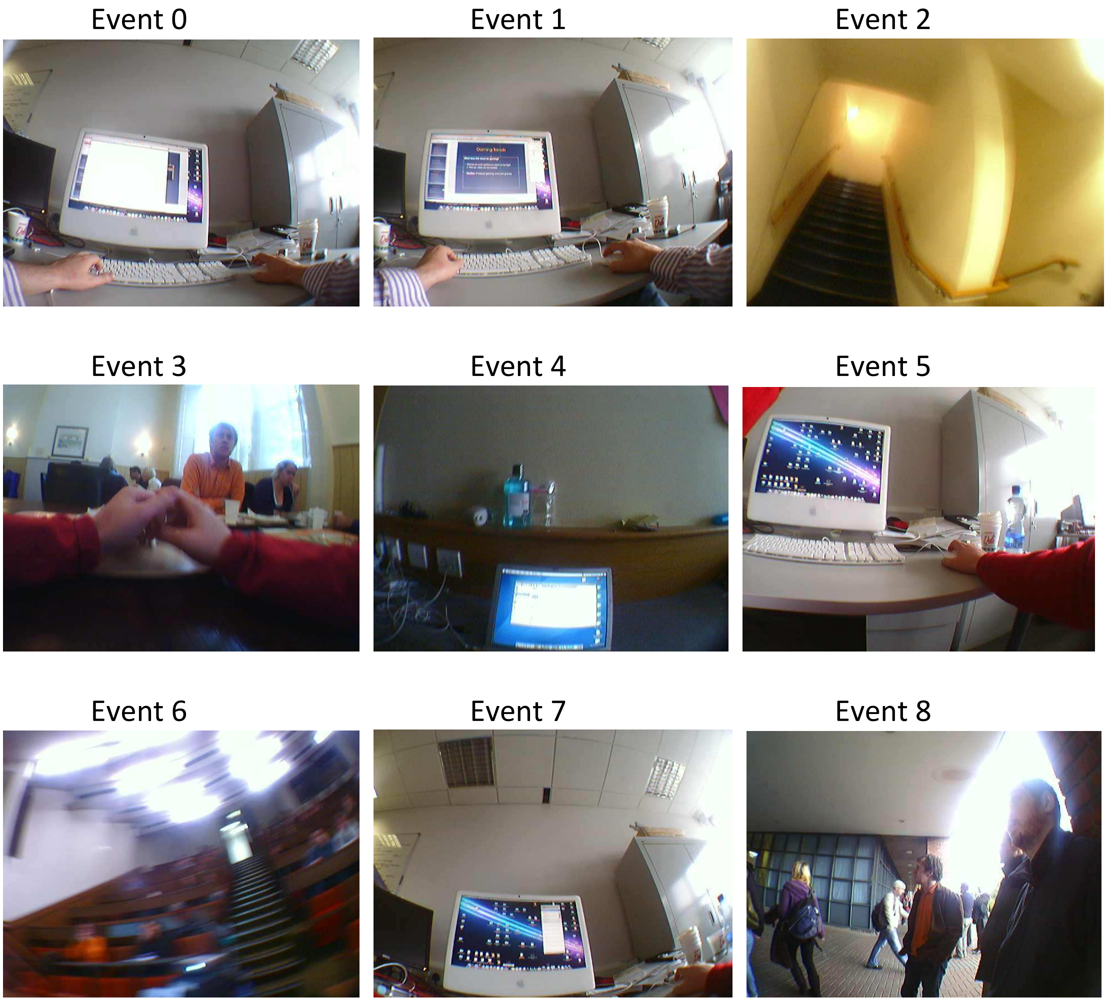
- **Saliency Measure**: here we measure the “different” from the image.

After the processing take part, the result values have to be normalized.[20]

There are two other values, automatically captured by the SenseCam that can help us to predict the quality present in the images taken.

We can relate to our image the values given by the accelerometer at the time the image was taken. If the accelerometer registered some kind of motion at that precise moment, it is very probable that the resultant image is blurred. Besides, many pictures suffers from lack of light or from excess of brightness due to ambient light, so if we have a look at the values registered by the light sensor we can have an idea of the images that present a poor illumination.

In the next example we have chosen the middle image of the nine first events on the day of a wearer:



Event 0

Event 1

Event 2

Event 3

Event 4

Event 5

Event 6

Event 7

Event 8

Figure 2-20: Middle image of the nine first events in the wearer´s day.

As we can see in the previous figure, almost every image selected satisfies their role as a representative photo. However, we can see that there has been a mistake between event 0 and event 1, as they should take part of the same event. Besides, the central image of event 6 is perhaps too blurry to be a good representative. It is probably caused by a quick movement of the wearer. This is why we need to consider the quality of the image chosen to have good representatives.

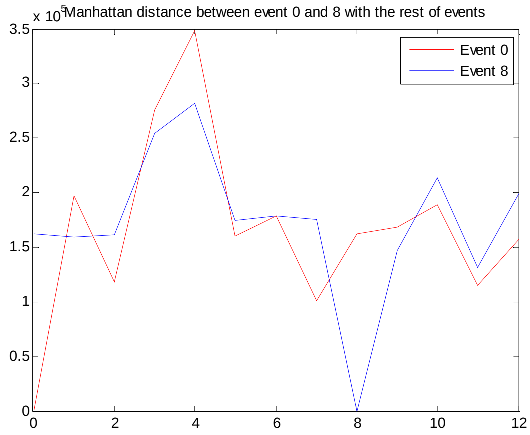


Figure 2-21: Manhattan distance between Event 0 and 8 of figure 2-20 with the first 12 events. (Using colorHistogram)

Once we have the quality score of the images, we can apply the Kapur[14] adaptative thresholding technique to know which of the images in an event cannot be selected as representatives due to their poor quality.

2.3.3.3 Calculating event novelty

The last step in the processing of the images is to calculate how important or how exceptional an event is. When the user wants to review his past activities, we can understand that it will be more preferable for him to take a look at those images of events that are more unique and unusual. It will be worthless to review those pictures in which the user is just sitting at his desk working, being more interesting to look through events like his daughter birthday party or meeting an old friend.

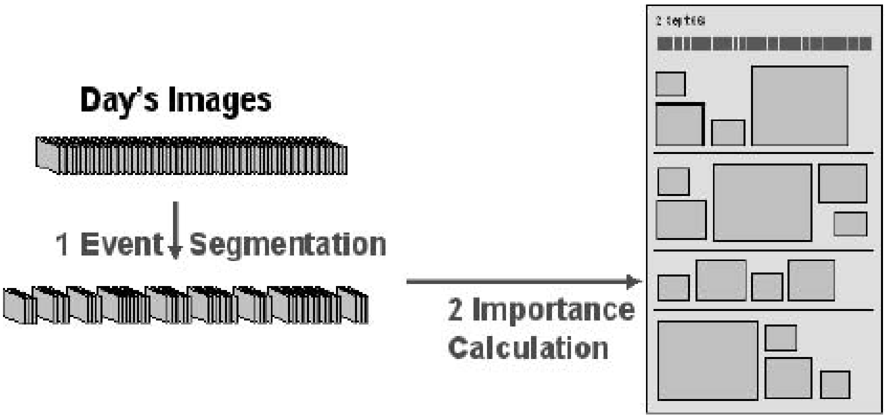


Figure 2-22: Determining event importance. [22]

Event representative vectors, composed by the low-level MPEG-7 visual features and accelerometer, light, passive infrared, and temperature values, are compared one another using the Manhattan distance to determine how similar they are. If an event contains a lot of uniformly dark images, the novelty score will be zero.

The novelty selection technique, basically consist in picking the most dissimilar event during a certain period of time. This period of time can vary. In [22] we can read a study about novelty calculation over different "windows": the previous 7 days, previous 3 days, previous 14 days, events within ±2 hours (of the event we are trying to compute the novelty of) from previous 7 days and same week day in the previous 21 days.

Although "previous 14 days" is the window that requires a higher computational cost, it is demonstrated that is the optimal one referring to the right amount of previous days to consider. However, an advantage of the 2h-time constrain approach is that the novelty of an event is calculated based on how different it is to all other events and also considering the time of day. This is important, because if we are sitting in front of the computer in the middle of the night, this event is more unique that if we are sitting in front of our computer during working hours.

The same day of week in the previous 21 days is the method with worst results, so a combination of "previous 14 days" and 2h- time constrain approach will be the best way to proceed.

Following these previous three steps: event segmentation, landmark photo selection and event novelty calculation, we finally have our photos segmented into events and ready for the user to review them in a clear and easy way:

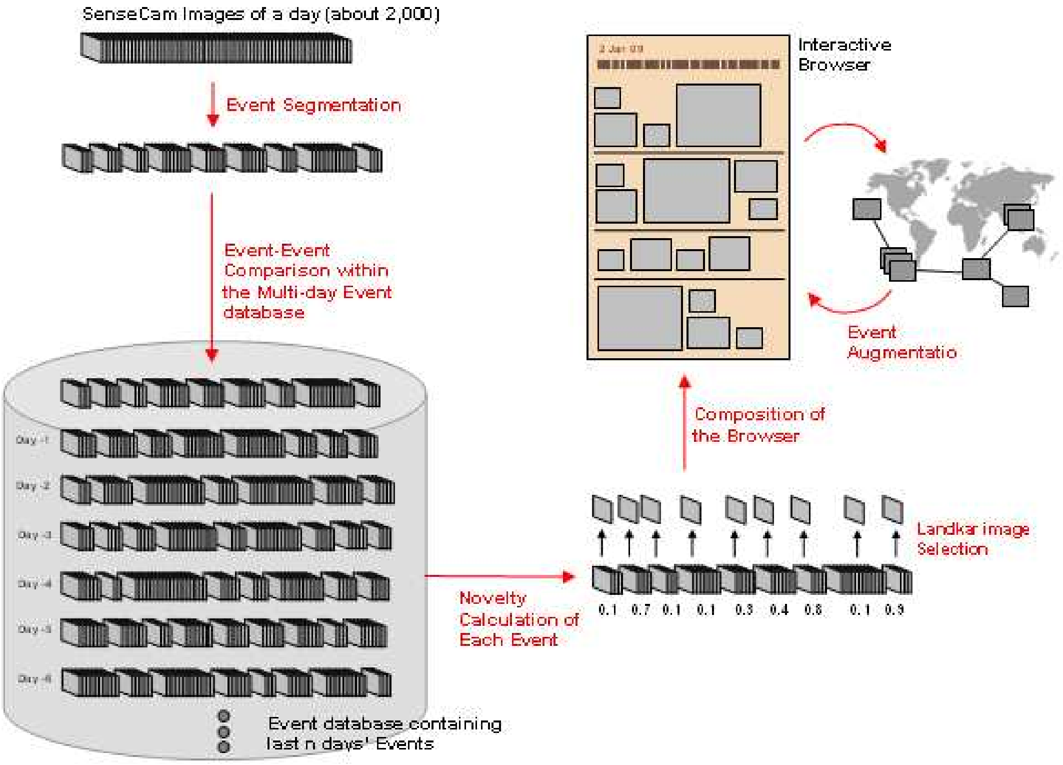

SenseCam Images of a day (about 2,000)
Event Segmentation
Event-Event Comparison within the Multi-day Event database
Day -1
Day -2
Day -3
Day -4
Day -5
Day -6
Event database containing last n days' Events
Novelty Calculation of Each Event
0.1 0.7 0.1 0.1 0.3 0.4 0.8 0.1 0.9
Landkar image Selection
Composition of the Browser
Interactive Browser
Event Augmentatio

Figure 2-23: Entire process of event segmentation, landmark detection and novelty calculation.

The main aim of our project is to make an interactive system to analyze dietry habits. Thus, our software has to be able to show the user the different images where the wearer is having lunch; breakfast or taking any food, discarding all other pictures where the activity is not related to the diet.

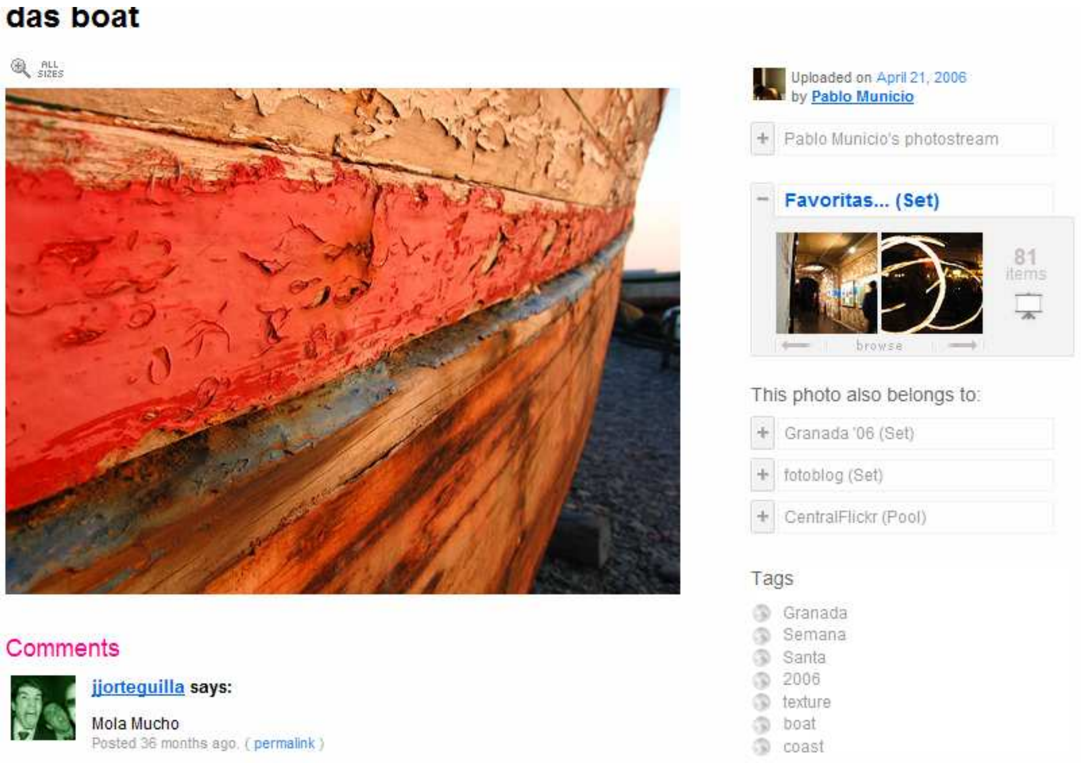
In order to do that, we have to process our images with the finality of "guessing" the activity which is taking place in each of the events. (Eating, riding a bicycle, outdoors images, indoors activities, sitting in front of the computer, watching television..)

This is called **concept detection** and we explain how it is done in the following section.

2.3.4 Concept detection

For a long time, searching for images based on text was the optimal method because the number of images was not large and because there were a limited number of people that needed to access huge images databases. However, the conditions don´t remain constant nowadays. We have sites in the Internet like Google picassa and Flickr that attract a large amount of visitor everyday and high-end imaging professional equipment are cheaper now. Nevertheless, looking for images is still a difficult task. For example, Flickr do not allow pictures to be accessed based on their content, but in the annotation that the user creates. This type of access will be sufficiently accurate if the annotations made by the user were perfect. Unfortunately this will never happen.

Therefore, we will have to employ content-based methods to access digitally stored images quick and without problems. A disadvantage of these content-based methods is that the computational cost is very high, and it will not be reasonable to make the computation each time we want to search for a photo, so we can automatically create textual labels based on the image´ content and do the search from them. [23]



das boat

ALL SIZES

Uploaded on April 21, 2006
by Pablo Municio

+ Pablo Municio's photostream

− Favoritas... (Set)

81 items

browse

This photo also belongs to:

+ Granada '06 (Set)

+ fotoblog (Set)

+ CentralFlickr (Pool)

Tags

Granada
Semana
Santa
2006
texture
boat
coast

Comments

jjorteguilla says:

Mola Mucho

Posted 36 months ago. (permalink)

Figure 2-24: Image from Flickr and the "Tags" to search the photo

To carry out with success many of SenseCams applications, this concept detection is basic. We have to take into account that the images will have poor resolution, they will be distorted due to the fisheye lens and some images will be more dark than desired because of the lack of flash. As we can see, concept detection is not an easy process. Besides the problems due to the characteristics of SenseCams, we will have to deal with other problems due to the following reasons:

- The number of positive samples is very small compared to the negative ones.
- Many objects are difficult to detect because two same types of objects can look very different to each other due to their position, illumination, appearance…[25]
- Some concepts are highly abstract and difficult to judge.

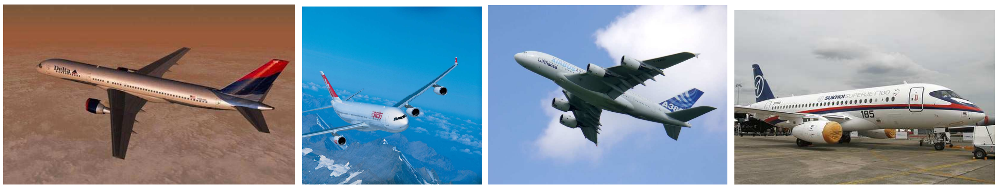


Figure 2-25: Different images of a same concept.

The concept detection process can be divided in three stages:

- Supervised learner:

  The detection of concepts is basically the recognition of a pattern. Given an image or a part of it, we have to measure how similar it is from different patterns, which will indicate if a concept is present in the image or not. For supervising learning of concepts, we use the Support Vector Machine (SVM). With SVM, we view input data as two sets of vectors in an n-dimensional space. Our method will construct a hyperplane that will maximize the margin between the two data sets. The larger this margin is, the lower the generalization error of the classifier.

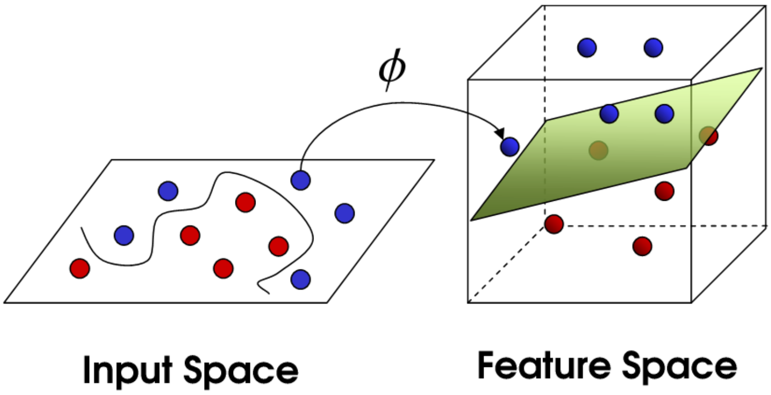


Figure 2-26: Pattern Classification by an SVM.

- Visual feature Extraction:

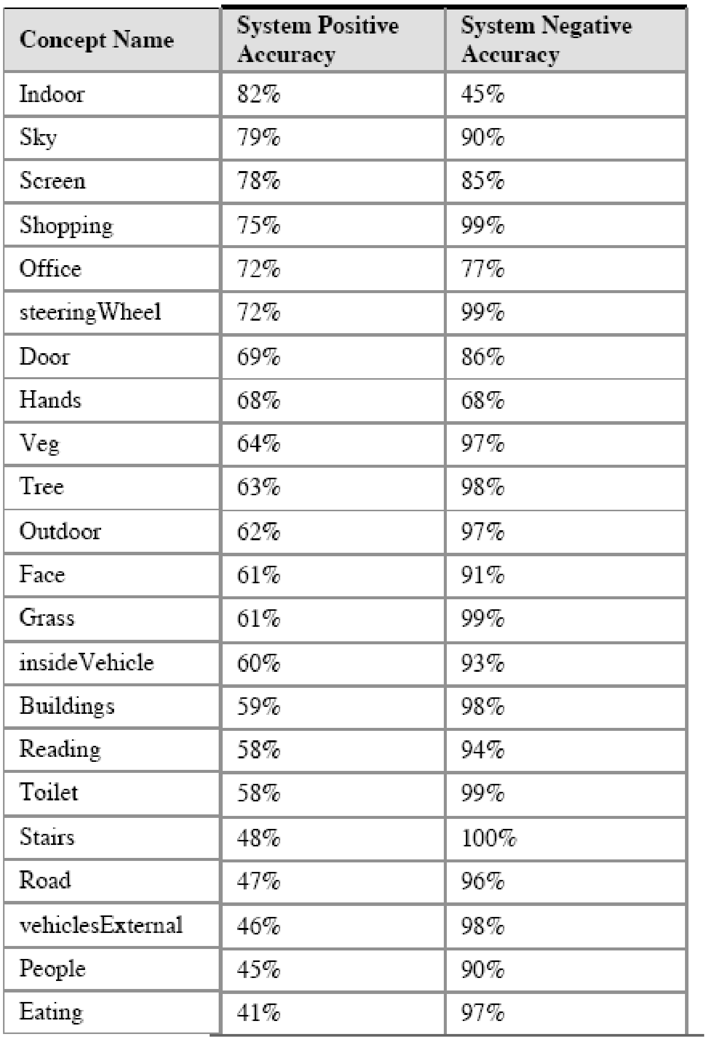
  Here, we adopt the codebook novel [24], which represents an image as a distribution over codewords. Following [25], we can build this distribution by dividing an image in several regions with rectangular shape. To obtain two separates codebook models, we will employ two methods in order to extract visual features (they measure colored texture):

  - Wiccest features: These features utilize natural image statistics to model texture information. Distributions of edges describe the texture in an image. These features are well suited to detect natural sceneries.

  - Gabor features: They are good to detect artificial constructions, as they respond to regular patterns in a given orientation on a given scale and frequency and they are also sensitive to color planes. [26]

- Feature and classifier fusion: here, we employ average fusion of classifier probability scores. After this, we obtain our final concept detection score.

Then, for each concept, we calculate a threshold value. We can use the Kapur automatic thresholding technique. This is an entropy based non-parametric technique that don´t require any training, so it can be easily applied to a large collection of images. All images above this threshold will be considered positives, and every image below will be a negative.



Concept Name	System Positive Accuracy	System Negative Accuracy
Indoor	82%	45%
Sky	79%	90%
Screen	78%	85%
Shopping	75%	99%
Office	72%	77%
steeringWheel	72%	99%
Door	69%	86%
Hands	68%	68%
Veg	64%	97%
Tree	63%	98%
Outdoor	62%	97%
Face	61%	91%
Grass	61%	99%
insideVehicle	60%	93%
Buildings	59%	98%
Reading	58%	94%
Toilet	58%	99%
Stairs	48%	100%
Road	47%	96%
vehiclesExternal	46%	98%
People	45%	90%
Eating	41%	97%

Table 2-7: Accuracy of detection for some concepts.

It was found that the accuracy of the system was more than acceptable. From the 95.907 images that are judged manually and with the automatic technique in [27], the 75% were correctly classified by the system. From them, the system correctly identified 57% of true positives and 93% of negatives.

3 Design

In this chapter, we will explain with detail the design of our interactive dietry habits analyzer and the evolution of its architecture during this project. A general block diagram will be shown and an overall description of each point will be presented.

3.1 Description of system architecture

Firstly, we are going to show a project schematic as we thought about it since the beginning, from the time a real user decides to monitor his diet using a Microsoft SenseCam till he receives diet personalized advices from our software. For a better and clearer understanding, we can divide it in three parts. The first and the third one will require the user intervention, and the second one will be done automatically.

3.1.1 First stage, taking pictures

This stage of the process is explained in Figure 3-1. A man, a woman or a child decides that it would be a good idea to improve his/her dietry habits using SenseCams (1). It can be because other used methods have already failed or just because nobody born knowing how to eat correctly and it´s never too late to learn. Besides, the model we are presenting requires little effort from the user, making it comfortable to use.

After placing a SenseCam around his neck, it starts taking pictures passively, without user interaction. Pictures will be taken each 30 second approximately or every time there is a significant change in the data that SenseCam sensors collect. Of course, if the user is in some kind of relevant situation, he can manually take a picture using the SenseCam photo trigger (2, 3). (See 2.2.1)

After a certain period of time, we have to upload the images taken using a personal computer. (4) We can upload the photos at the end of the day to let the second part of the process start. After uploading the images, the user cannot forget to charge the SenseCam battery to have it ready to for next day.

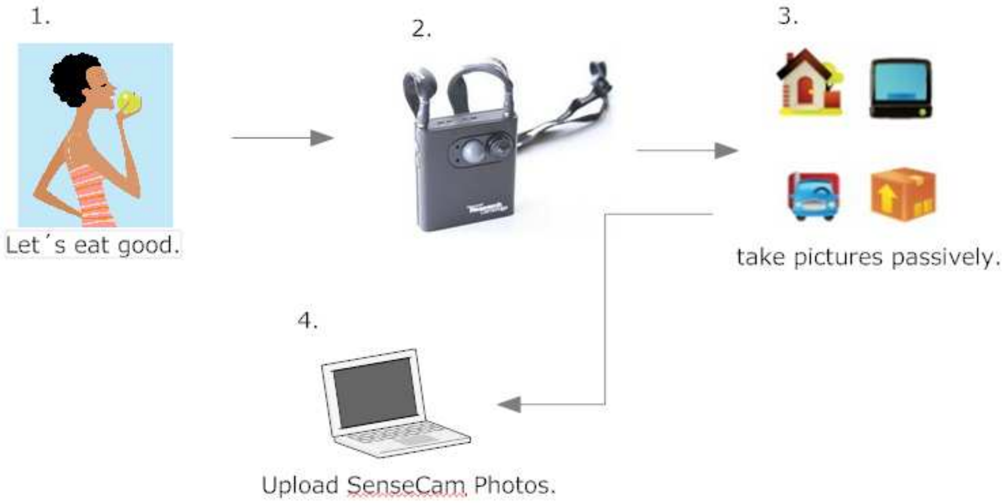


Figure 3-1: First stage

It is important to install the “Dietry Habits Analizer” software in our personal computer. This is the tool that we will use to review all our eating images and to introduce all the additional data needed. Besides, it will give us personalized advices about our diet.

3.1.2 Second stage, image processing

This stage is done automatically, without user intervention. Our purpose is to prepare and organize all data extracted from the images. When the software needs to retrieve any data, it could find it just connecting to the database and searching for the field needed.

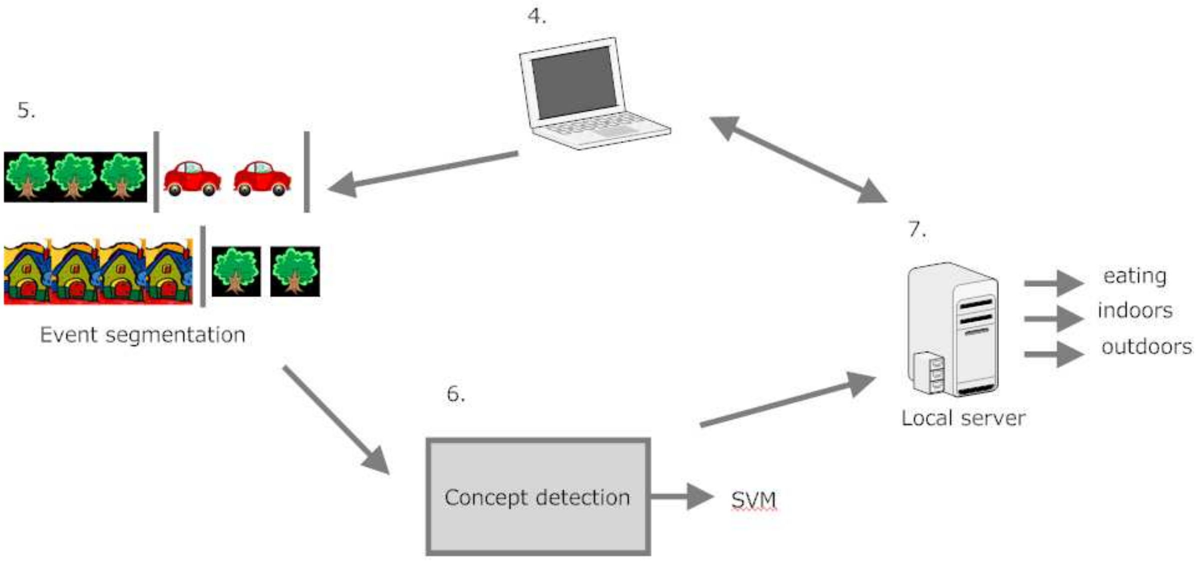


Figure 3-2: Second stage.

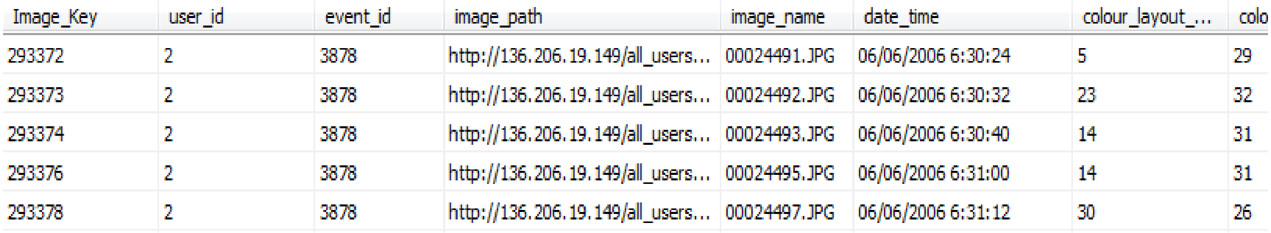
Once the wearer has uploaded the images using a computer (4), the event segmentation and concept detection software will do their work (5, 6). Due to practical reasons, we have only used the data provided by the sensors installed in the SenseCam to segment all the events. This is because the computational cost is too much lower and the accuracy in the segmentation is only a 3.3% better including MPEG-7 sources.

As a result, all data extracted from our images are perfectly ordered in a database. (7).The dietry monitor software that we have installed in our computer is now able to read the database and extract all the information needed to analyze and show it to the user (in our case it is especially important the probability of the wearer taking any kind of food).

The way the data is organized is very intuitive. Each row of information is composed by several fields:

- Key number and a “name of image” that indentifies it.
- Number that identifies the event.
- A number that identifies the user.
- The date and time of the shot.
- Visual features MPEG-7 vectors: color layout, color structure, edge histogram and scalable color.

- Accelerometer values that the SenseCam recorded in that instant of time, from each of the axis: x, y and z.
- Ambient temperature and values of the rest of sensors: light and passive infrared one.
- Trigger information. A “P” means that it was triggered by a change in passive infrared data, an “L” by a change in Light sensor data, an “M” means that it was manually triggered and a “T” means that the picture was taken after a certain period of time.
- Likelihood of each of the concepts taken place in each of the images: buildings, face, hands, eating…

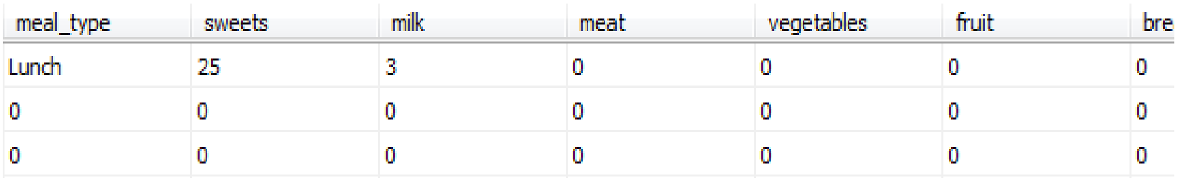


Image_Key	user_id	event_id	image_path	image_name	date_time	colour_layout_...	colo
293372	2	3878	http://136.206.19.149/all_users...	00024491.JPG	06/06/2006 6:30:24	5	29
293373	2	3878	http://136.206.19.149/all_users...	00024492.JPG	06/06/2006 6:30:32	23	32
293374	2	3878	http://136.206.19.149/all_users...	00024493.JPG	06/06/2006 6:30:40	14	31
293376	2	3878	http://136.206.19.149/all_users...	00024495.JPG	06/06/2006 6:31:00	14	31
293378	2	3878	http://136.206.19.149/all_users...	00024497.JPG	06/06/2006 6:31:12	30	26

Figure 3-3: Fields of image information.

For the correct operation of our software, we have added some fields to the database related to the meals taken and the type of food ingested:

- Type of meal: Breakfast, lunch, afternoon snack, dinner or extra meal.
- How much food is taken from each of the six groups that form the food pyramid:
    - Sweets and oil.
    - Milk.
    - Meat, fish and eggs.
    - Vegetables.
    - Fruit.
    - Bread and cereals



meal_type	sweets	milk	meat	vegetables	fruit	bre
Lunch	25	3	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

Figure 3-4: Extra fields added for our special purpose.

Once everything is stored, the user can start interacting with the graphical interface. Pictures about his diet will be shown, and data about the type of meal and the quantities taken will be introduced.

It is time now of receiving some advises based in the particular dietry habits that the user has.

3.1.3 Third stage, interaction

In this part of the process, the time of the real purpose of the study has arrived. Here, the user will interact with the software and receive some custom-made advices. After step one and two, we now have all the images in our laptop and all the visual information and data collected by sensors is well organized.

Now, the software created for this project will be launched and due to its friendly and easy interface, the user will interact with it without reading any special manual: all the information required is right in front of him. After introducing the information needed for the software to work, some advices will be created regarding the diet that the wearer has had in the days selected. (See Development)

The diagram showed below is a continuation of figures 3-1 and 3-2. As we can see in figure 3-5, the user has to introduce some of his diet information (8) using the graphic interface. This data will be saved in the database and afterwards, it will be read and analyzed to create good dietry advices. Finally, it will be up to the user to follow them!!(10)

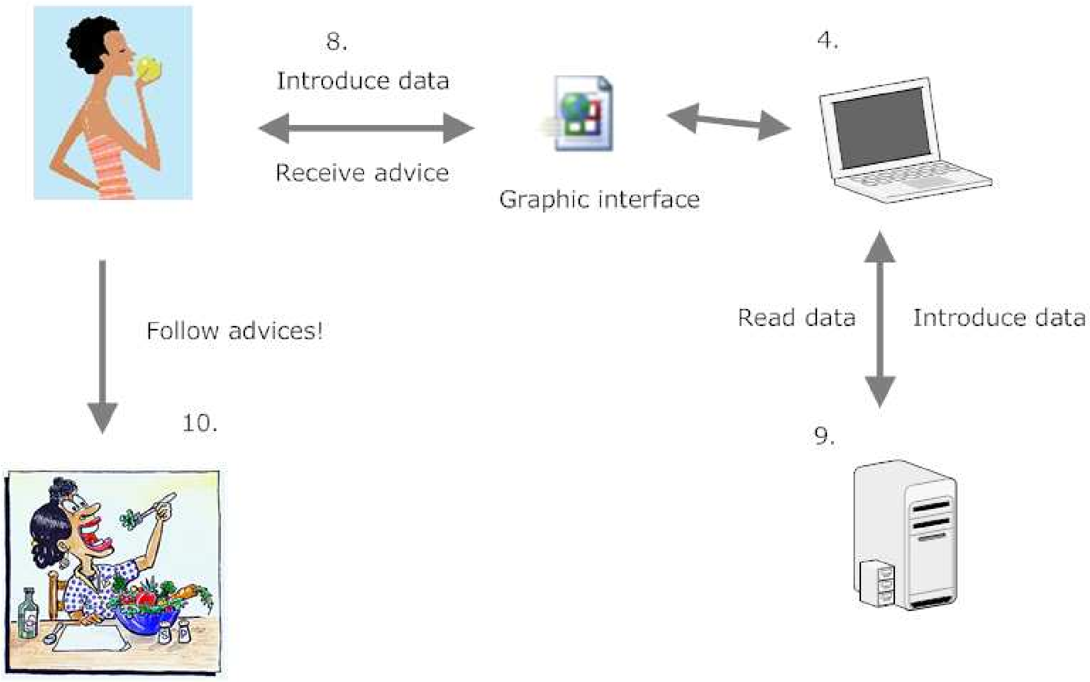


Figure 3-5: Third step: interaction with the user.

4 Development

4.1 Introduction

Now we are going to briefly comment all the technologies that are part of our design and the use that we have done of each of them.

Also, we are going to introduce all the problems presented during the development of the design proposed in 3.Design and how the appropriate solution was found.

4.2 Microsoft Visual Studio 2005

Microsoft Visual Studio 2005 is an Integrated Development Environment (IDE) from Microsoft. This is the one chosen for the development of this project because it has many advantages for creating robust and elegant applications using ASP.NET.

Amongst other advantages, it has code completion, which means fewer errors and less typing, an HTML editor which provides design and HTML views that update each other in real time and an integrated debugger, which allows to follow the code step by step making easy to find the functional errors.

4.2.1 ASP.NET

ASP.NET is a web application framework developed and marketed by Microsoft. With this tool, we can build dynamic web sites, web applications and web services. We can write ASP.NET code using any .NET languages (C++, J#, C#...). In the case of the present project, we are going to use C# for the code behind.

Web forms are the main building block for application development. They are contained in files with an ".aspx" extension. These files contain Hipertext Markup Language and with it we will model the look of the web page the way we want. Besides, in this same page we will have javascript language, which will be very helpful with some of the issues of the design.

In our project, we have two web forms: one for the principal page and another one for the advice page. Each of them has a "code-behind" file. Both are written in C# and they show how the program responds to different events, for example, a page being loaded or a control being clicked. [28]

4.2.1.1 C#

C- Sharp is a simple, modern, object-oriented programming language. It has its roots in the C family of languages and it also looks very similar to Java. C# is standardized by ECMA International as the ECMA-334 standard and by ISO/EC as the ISO/EC 23279 standard.

These are some of the design goals listed in the Ecma standard:

- Simple, moder, general-purpose, object-oriented programming language.

- Support for internationalization.
- Source code and programmer portability.

C#'s principal designer and lead architect at Microsoft is Anders Hejlsberg, who previously was involved in the design of Turbo Pascal, CodeGear Delphi and Visual J++.

With this language, we will develop the code-behind in our project, where we specify the action taken when any event occurs and where we will connect with the database to introduce and extract the information needed.

4.2.1.2 HTML

Hipertext Markup Language is a language for describing web pages. It allows creating structured documents by denoting structural semantics for text such as headings, paragraphs, lists, html images…

It is not a language itself; it is a markup language, which is a set of markup tags. These markup tags are used by HTML to describe web pages. These tags are surrounded by angle brackets like <html> and normally come in pairs like <b> and </b>.

The basic structures for HTML markup are elements. Elements have two basic properties: attributes and content.

4.2.1.3 Javascript

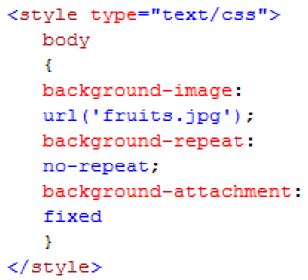
Javascript is a scripting language, that is, a programming language that allows control of one or more software applications. It is used in the form of client-side Javascript for the development of dynamic websites. Javascript supports all the structured programming syntax in C: if, while, for…

Javascript was designed to look like Java, but to be easier for non-programmers to work with. However, it is unrelated to the Java programming language, even though both of them have some similarities: they both have C influences and JavaScript copies many Java names and naming conventions.

It was originally developed by Brenda Eich of Netscape under the name Mocha. Later, the name was changed to LiveScript to end up being named JavaScript. “Javascript” is a trademark of Sun Microsystems.

4.2.1.4 Css

Casvading Style Sheets is a style sheet language used to describe the presentation of a document that is written in a markup language. In our project, we use Css to present our pages with special images in the background:



```
<style type="text/css">
   body
   {
   background-image:
   url('fruits.jpg');
   background-repeat:
   no-repeat;
   background-attachment:
   fixed
   }
</style>
```

Figure 4-1: Use of Css in the present project.

4.3 Microsoft Sql Server

Microsoft Sql Server is a relational database model server produced by Microsoft. The model for a relational database is based on first-order predicate logic. The main unit of data storage is a database: a collection of tables with typed columns.

To retrieve data from an SQL Server database, we will have to query it. The query is expressed using a variant of SQL called T-SQL. This way we specify what is to be retrieved, and it will be a task for the query processor to figure out the sequence of steps that will be necessary to retrieve the needed data.

SQL Server has been evolving since the late 1980s. Microsoft SQL Server was originated as Sybase SQL Server in 1987. The version used in this project: Microsoft SQL Server 2005 was released on November 7th 2005.

Along this project, we have used Microsoft Sql Server to retrieve needed data in several occasions which will be explained along this document. We created a database (See 3.1.2) with all the data extracted from the images and from the SenseCam sensors. Besides, we used stored procedures when our system required the execution of several SQL statements. This way, our application had to call only the stored procedure. We have implemented two stored procedures, one for getting the days available: days when we took photographs and other to get the images where the user is eating.

4.3.1 Stored procedures

In the following figure we can see an overall view of how the first part of the development of the system works:

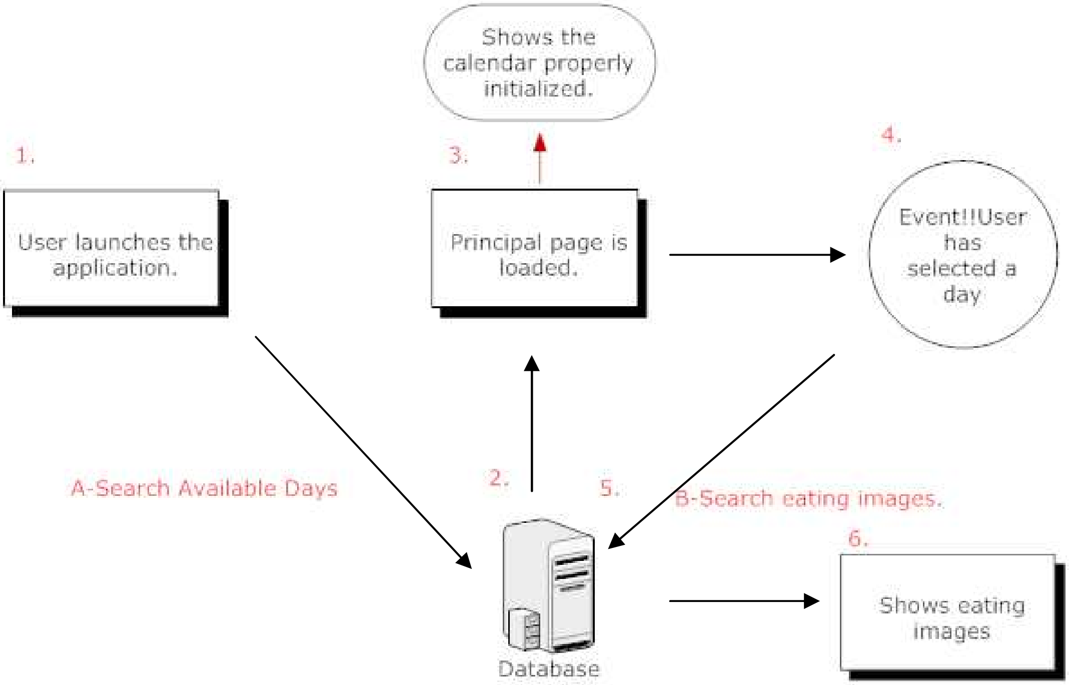
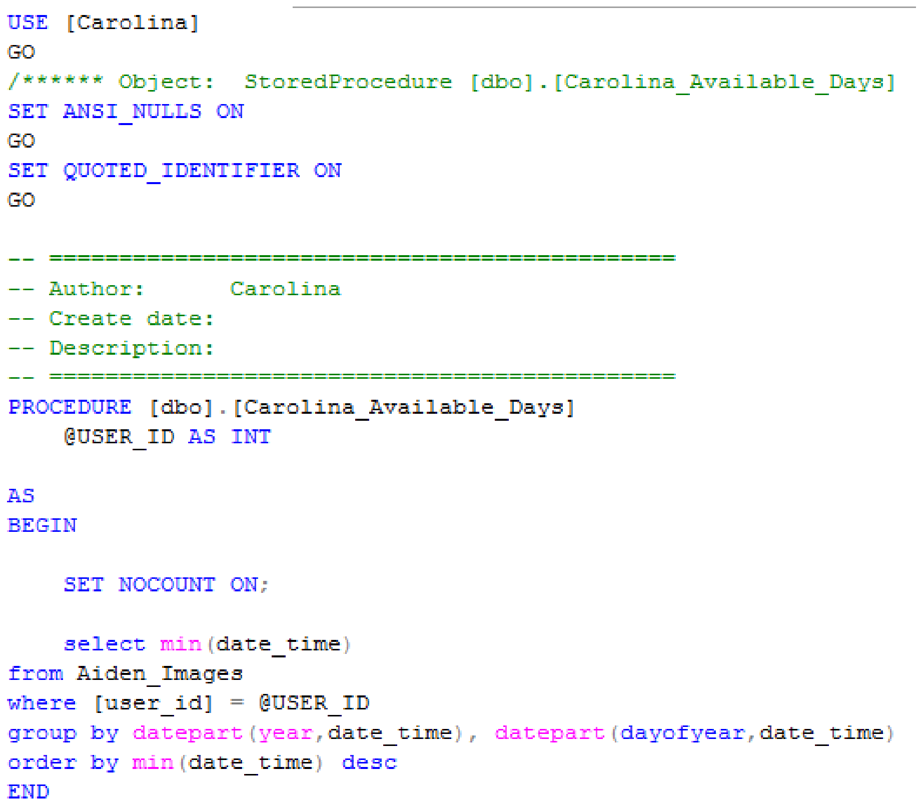

Shows the calendar properly initialized.
1.
3.
4.
User launches the application.
Principal page is loaded.
Event!!User has selected a day
A-Search Available Days
2.
5.
B-Search eating images.
6.
Database
Shows eating images

Figure 4-2: Application´s start

The steps where the stored procedures are called are marked in red. The first stored procedure is called when the user launches the application and it is necessary to search the available days to initialize the calendar. When the pages loads, we would see a calendar opened in the day when we took the first picture. Then, the user must choose a day in the calendar and the second stored procedure will be called. The application will show then the images where the user is eating.

4.3.1.1 Available days

The first stored procedure implemented looks for those days in the database that have information. For that, we have to examine the images´ timestamps to realize how many days of information do we have. To work, it needs a parameter: the identification of the user. If the identification number is correct, the stored procedure will return the information needed.



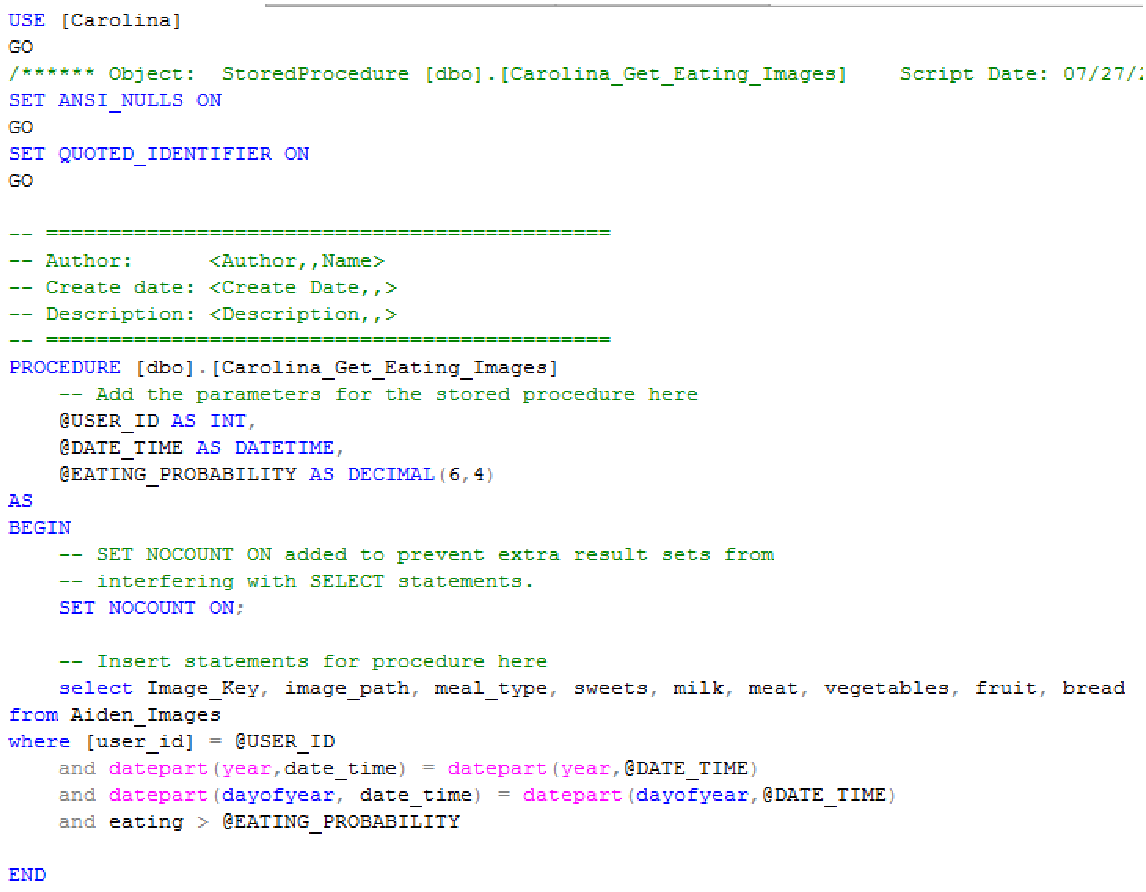
```
USE [Carolina]
GO
/****** Object:  StoredProcedure [dbo].[Carolina_Available_Days]
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

-- =============================================
-- Author:      Carolina
-- Create date:
-- Description:
-- =============================================
PROCEDURE [dbo].[Carolina_Available_Days]
    @USER_ID AS INT

AS
BEGIN

    SET NOCOUNT ON;

    select min(date_time)
from Aiden_Images
where [user_id] = @USER_ID
group by datepart(year,date_time), datepart(dayofyear,date_time)
order by min(date_time) desc
END
```

Figure 4-3: First stored procedure: Available days.

As we can see in the figure above, if the identification number is introduced correctly, the time stamp of the first image in each different day is selected.
This way we can know which days we have recorded data.

4.3.1.2 Eating images

This is the second stored procedure created. When the user presses a date in the calendar, the application must show him the images where he is taking any food. When we implemented concept detection (See 2.3.4), each image was scored with the likelihood of a certain concept taking place. This way, we have in the database the probability that “grass”, “hands”, “buildings”, “eating” and many others concepts are present in each image. So, to find which are the images where the user is eating, we just have to read the likelihood of “eating” and choose those images where the probability is high enough. Of course, concept detection is not an exact science, and it is possible that some of the images chosen as eating weren´t eating images actually. We have solved this problem allowing the user to manually “delete” these incorrect shown images. Of course, instead of deleting the image, what the user is really doing in changing the eating probability to “0”.



```
USE [Carolina]
GO
/****** Object:  StoredProcedure [dbo].[Carolina_Get_Eating_Images]    Script Date: 07/27/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

-- =============================================
-- Author:      <Author,,Name>
-- Create date: <Create Date,,>
-- Description: <Description,,>
-- =============================================
PROCEDURE [dbo].[Carolina_Get_Eating_Images]
    -- Add the parameters for the stored procedure here
    @USER_ID AS INT,
    @DATE_TIME AS DATETIME,
    @EATING_PROBABILITY AS DECIMAL(6,4)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Insert statements for procedure here
    select Image_Key, image_path, meal_type, sweets, milk, meat, vegetables, fruit, bread
from Aiden_Images
where [user_id] = @USER_ID
    and datepart(year,date_time) = datepart(year,@DATE_TIME)
    and datepart(dayofyear, date_time) = datepart(dayofyear,@DATE_TIME)
    and eating > @EATING_PROBABILITY

END
```

Figure 4-4: Second stored procedure: Eating images.

The parameters needed to make this stored procedure work are the user identification, the date and the eating probability. For a given day, we will select the images with higher probability of eating occurrence than the probability introduced as a parameter.

In this project we will use a probability of 0.4. This way, we will show in the screen all those images where the probability of eating occurrence is higher than 0.4 organized by the time the SenseCam took the picture.

If we analyze the text in figure 4-4, we can see that besides image_path, more fields are selected. These fields contain the information about the meal that is taking place in the picture (breakfast, lunch, dinner…) and which are the nourishment that the wearer is having. These fields have to be filled by the user, but this procedure will be detailed later on.

4.4 Graphic interface

The main goal when designing and developing this application was creating a graphic interface which was easy to interact with regardless of user's expertise.

Actually, the graphic interface is the way the user has “contact” with all the advantages that the application has to offer, so it has to be friendly and with all the information necessary in full view of the user.

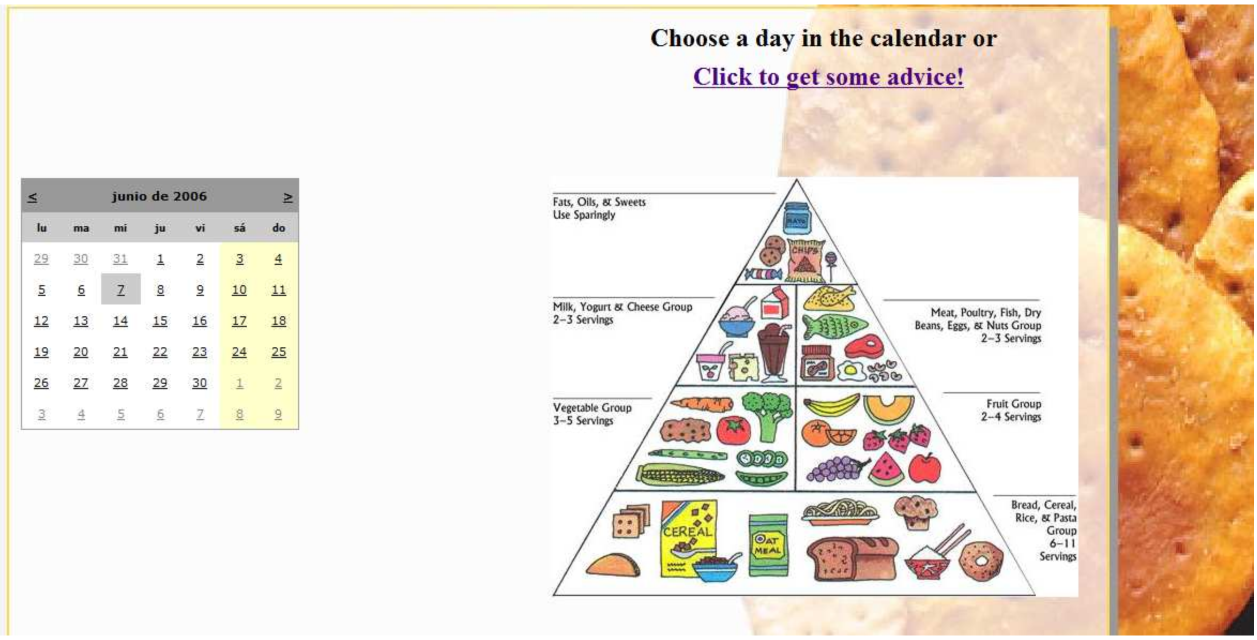
Besides, a usable and accessible design would help users interact with the application without having to read a user manual.

In the present project we needed to present several pages to the user, at least one for the main page and another one for the page where the advices are shown.

For this, we used two different web forms. An additional web form was used to show a slider, but we will explain this later.

4.4.1 Principal interface

We are going to consider as principal interface the first of the three web forms that we can find in this project. When the user launches the application, the calendar is initialized and the last day in which we have recorded images is marked (See 4.3.1). Next to the calendar, we present an image of a food pyramid. In this pyramid, the user can see the six different types of groups in which the food is divided and a brief explanation of the correct amount of serves per day for an adult user. At this stage, the user can decide if he wants to review the eating images of a certain day and introduce the eating information or if on the contrary he prefers to get some advice based on the amount and type of food taken a certain day. (See 4.4.2)



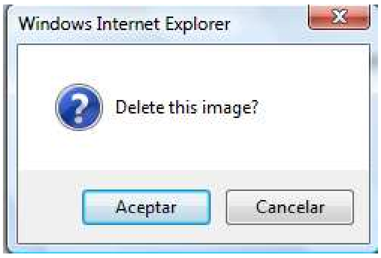
Choose a day in the calendar or

Click to get some advice!

≤			junio de 2006			≥
lu	ma	mi	ju	vi	sá	do
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

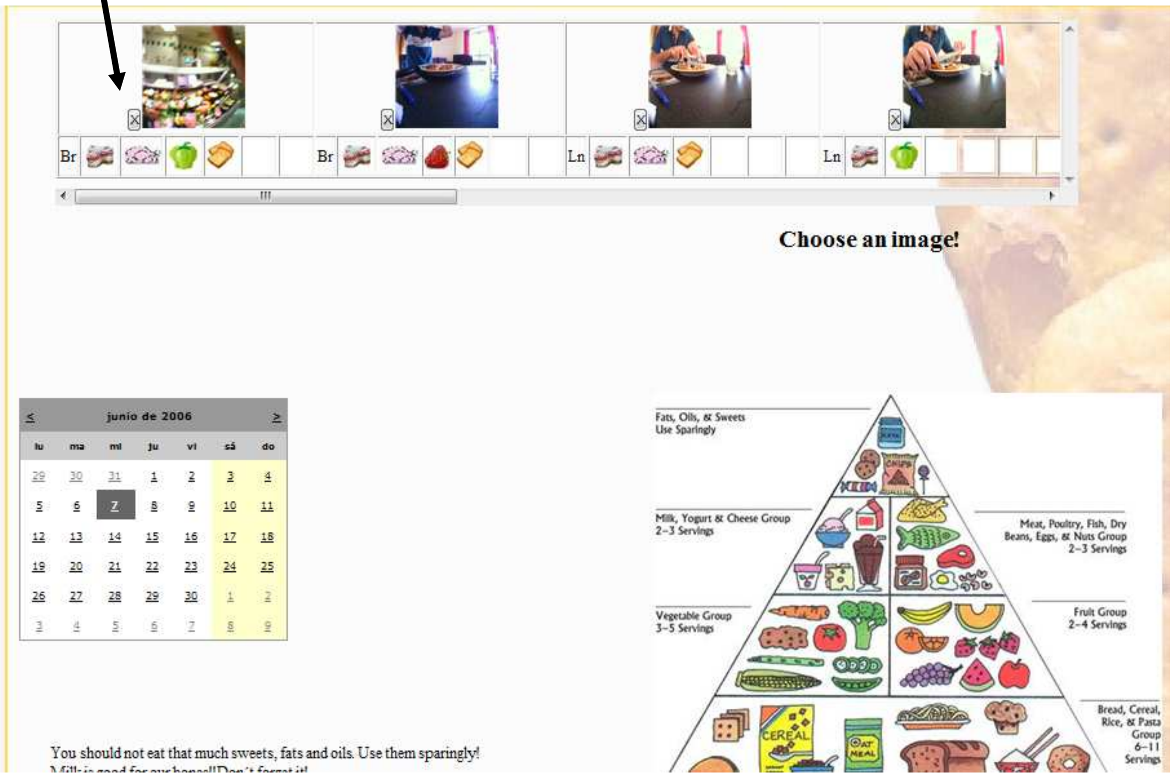
Figure 4-5: Principal page of the application.

If the user decides that he wants to revise the photographs taken a certain day, he just has to select a day and follow the instructions that will appear in the screen. As we can see in figure 4-6, all the eating images are shown with little icons which indicate what food is already annotated for that image in the database. If one of the images is not an eating image, the probability of “eating” will be changed to “0” just pressing the X beside it. This way, next time that we select that day; this image will not appear anymore. (Figure 4-6)



Delete this image?

Figure 4-6: Javascript popup after clicking X from erroneous image.


Br
Br
Ln
Ln
Fats, Oils, & Sweets Use Sparingly
Milk, Yogurt & Cheese Group 2–3 Servings
Meat, Poultry, Fish, Dry Beans, Eggs, & Nuts Group 2–3 Servings
Vegetable Group 3–5 Servings
Fruit Group 2–4 Servings
Bread, Cereal, Rice, & Pasta Group 6–11 Servings

Choose an image!

< junio de 2006 >

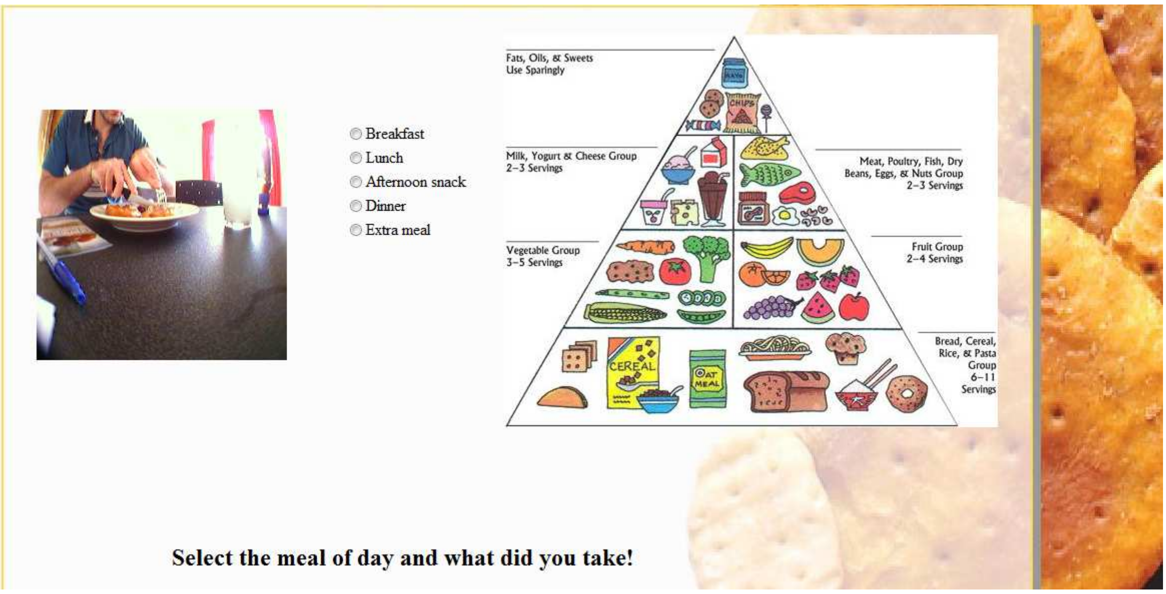
lu	ma	mi	ju	vi	sá	do
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

You should not eat that much sweets, fats and oils. Use them sparingly!
Milk is good for our bones!!Don't forget it!

Figure 4-7: Eating images of a day.

At this point, the user reads the screen and finds out that he has to choose an image. The image chosen has to be one with incomplete eating information, maybe because we never introduced any data on it or because we forgot to introduce something. All the information that we need to introduce in all the pictures of a day can be introduced by selecting every image one by one.

After selecting a certain image, the application will show us a larger version of the image selected. This way we will see every detail clearly. Now, we have to select the meal of day and what did we take. Thereupon, we have to introduce the type of food and the quantity taken. The pyramid beside the photograph looks like a normal image…but it is an html image. We can introduce the food just pressing in the corresponding place of the pyramid.


○ Breakfast
○ Lunch
○ Afternoon snack
○ Dinner
○ Extra meal
Fats, Oils, & Sweets Use Sparingly
Milk, Yogurt & Cheese Group 2–3 Servings
Meat, Poultry, Fish, Dry Beans, Eggs, & Nuts Group 2–3 Servings
Vegetable Group 3–5 Servings
Fruit Group 2–4 Servings
Bread, Cereal, Rice, & Pasta Group 6–11 Servings
CEREAL
OAT MEAL
CHIPS

Select the meal of day and what did you take!

Figure 4-8: introducing data.

When a food is pressed, a Javascript function will open the second web form: a slider. With this slider we can introduce how many serves we took of the food selected. (**See Appendix B**)

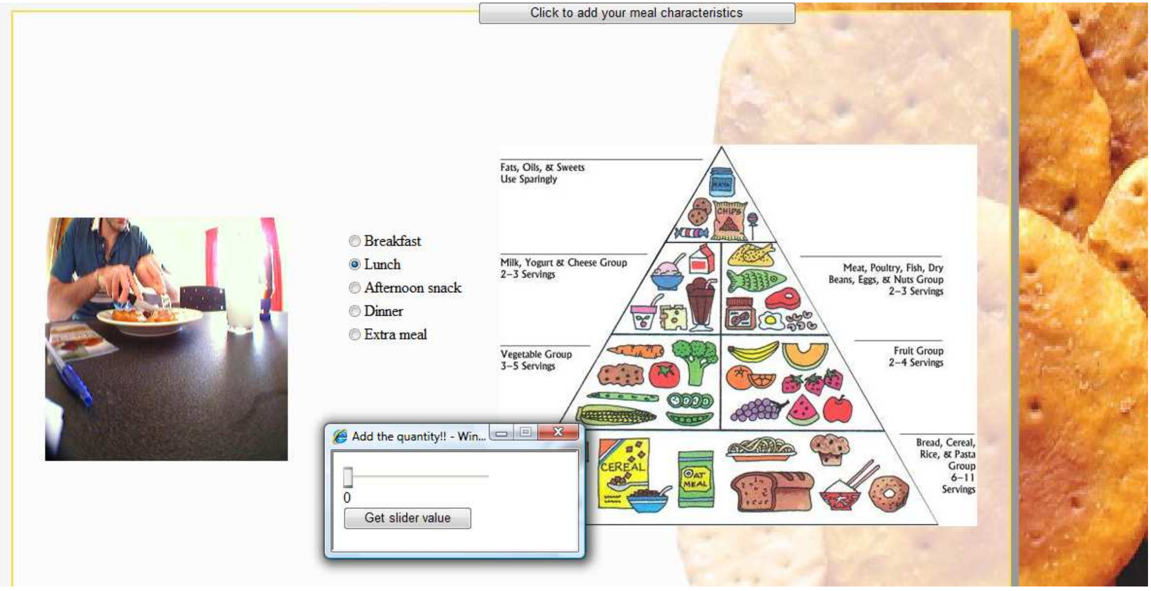

Click to add your meal characteristics
○ Breakfast
● Lunch
○ Afternoon snack
○ Dinner
○ Extra meal
Fats, Oils, & Sweets Use Sparingly
Milk, Yogurt & Cheese Group 2–3 Servings
Meat, Poultry, Fish, Dry Beans, Eggs, & Nuts Group 2–3 Servings
Vegetable Group 3–5 Servings
Fruit Group 2–4 Servings
Bread, Cereal, Rice, & Pasta Group 6–11 Servings
Add the quantity!! - Win...
0
Get slider value

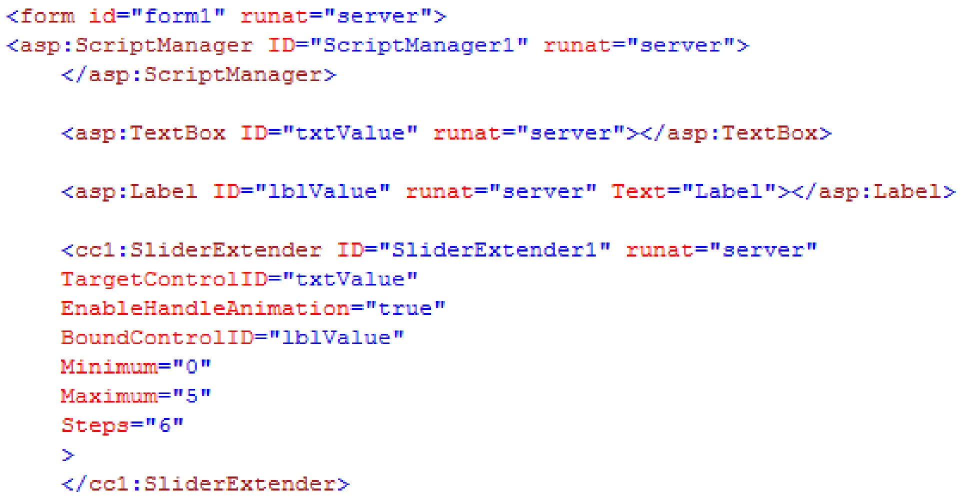
Figure 4-9: Slider to introduce quantities.

4.4.1.1 The slider

The slider is made using the second web form. This web form is opened with a Javascript function and it shares location with the first web form, as we can see in figure 4-9. To make it we used the **AJAX Control Toolkit**. This is an open-source project built on top of the Microsoft ASP.NET AJAX framework. This toolkit contains a lot of controls that make possible to easily create new controls from existing ones. In this case, we create a slider from an asp TextBox and a label. The TextBox will show the quantity selected and the label will contain the slider itself.

The values that can be chosen go from 0 to 5. They reference the serves taken of each food. To know how much is a serve; see (Appendix B)

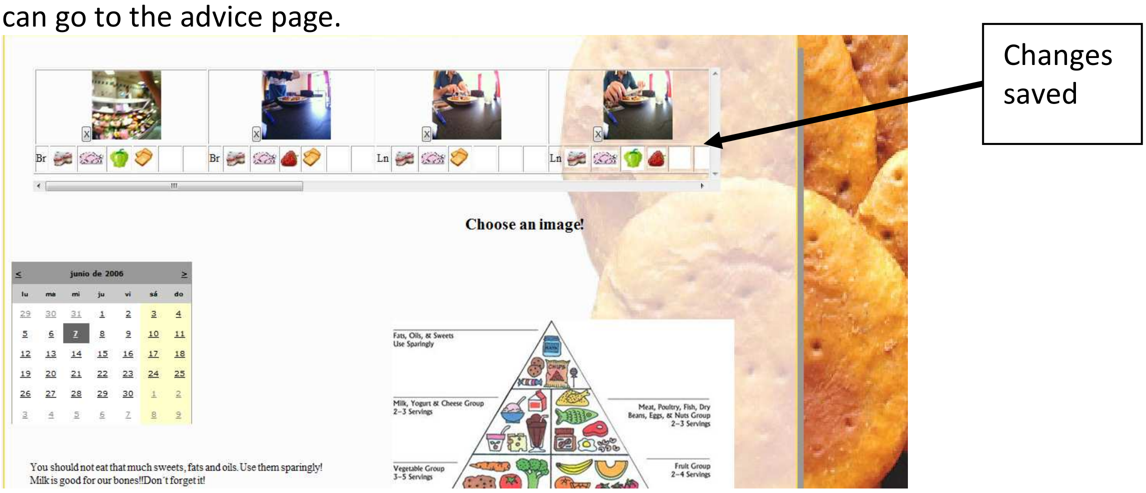
These values are recorded to count the serves taken from each food, each day. Then, they will be compared with guidelines of the optimal amount of food for each meal and for an entire day to see what the mistakes made are.



```
<form id="form1" runat="server">
<asp:ScriptManager ID="ScriptManager1" runat="server">
    </asp:ScriptManager>

    <asp:TextBox ID="txtValue" runat="server"></asp:TextBox>

    <asp:Label ID="lblValue" runat="server" Text="Label"></asp:Label>

    <cc1:SliderExtender ID="SliderExtender1" runat="server"
    TargetControlID="txtValue"
    EnableHandleAnimation="true"
    BoundControlID="lblValue"
    Minimum="0"
    Maximum="5"
    Steps="6"
    >
    </cc1:SliderExtender>
```

Figure 4-10: Slide using AJAX Control Toolkit

Once the data for an image is introduced, we can save the changes made and continue with other images. If we refresh the page we can see that the icons for the new food annotated appear under their picture. If we have finished introducing new data, we can go to the advice page.



Choose an image!

<			junio de 2006			>
lu	ma	mi	ju	vi	sá	do
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

You should not eat that much sweets, fats and oils. Use them sparingly!
Milk is good for our bones!!Don´t forget it!

Figure 4-11: Changes saved.

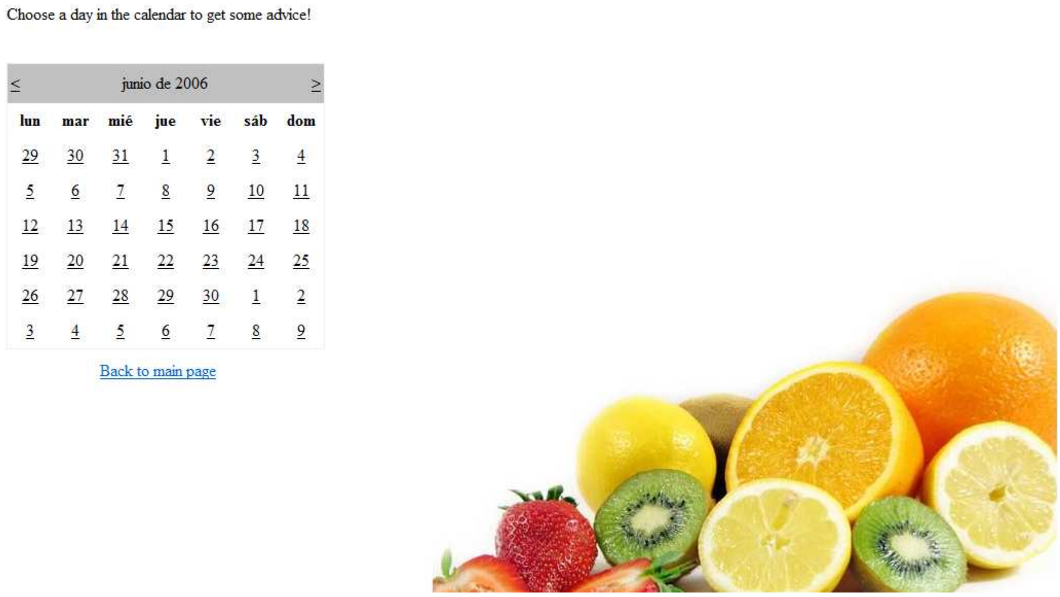
4.4.2 Advice graphic interface

In this third web form, we present the user an easy way to look for some advices based in what nourishments he took in any day he choose. The way it works is very easy for the user: he just has to select a day in the calendar and the system will show the eating images of that day and the type of food that he was taking in each one. This information has to be previously introduced by him.

Thereupon, a collection of advices appear on the screen. On top of the screen we can read overall advices of that particular day, especially those things that the user has to correct in the future (See figure 4-7). If the user took too much sweet during the day or didn´t take enough fruit, the application will advise him and will give some reasons of why it is important to take certain nourishment.

At the bottom of the page we can find more advices based on the analysis of each particular meal: Breakfast, Lunch, Afternoon Snack, Dinner and Extra Meal. (See figure 4-8)

In the “Advise” web page, we can find a link that guides us to the principal page. In case the user notices that he left a picture without information, he can easily go back to introduce it properly.



Choose a day in the calendar to get some advice!

< junio de 2006 >

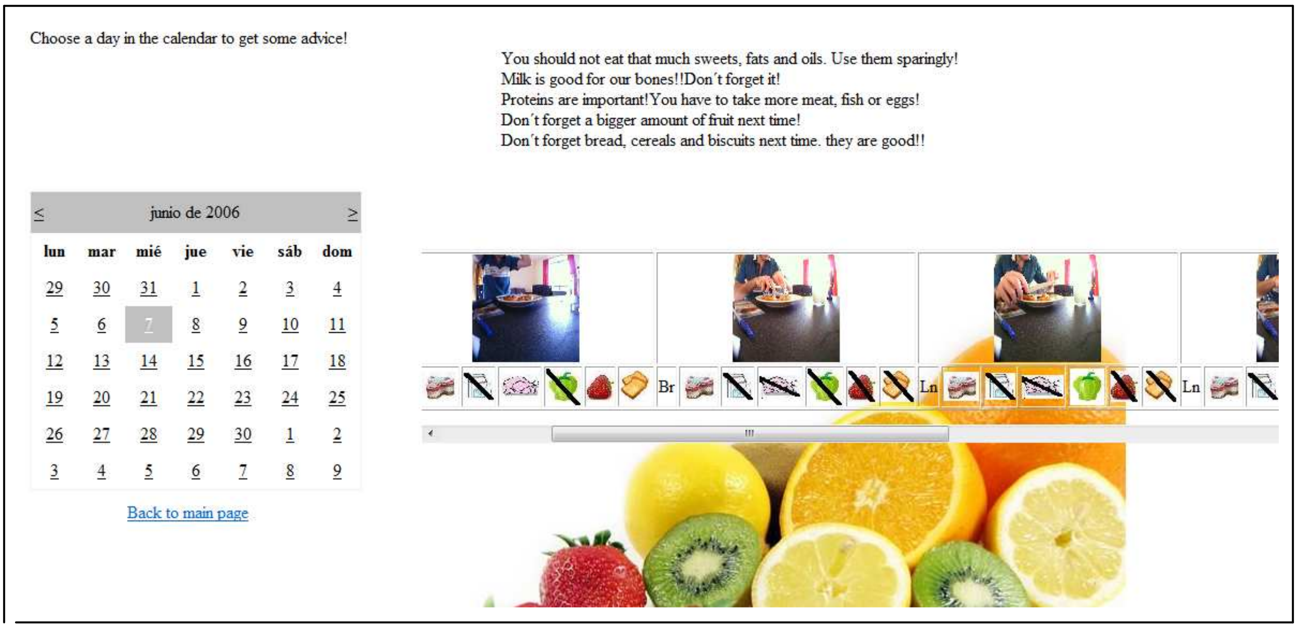
lun	mar	mié	jue	vie	sáb	dom
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

Back to main page

Figure 4-6: Advice page before selecting a day in the calendar.

The background picture that appears in figures 4-5 and 4-6 is a fixed image that remains motionless even though we play with the page, press a button or move the page to read the advices.

They are introduced using css language. These particular lines of code appear in the files with the .aspx extension. In these pages we also have the Javascript code and the html code.



Choose a day in the calendar to get some advice!

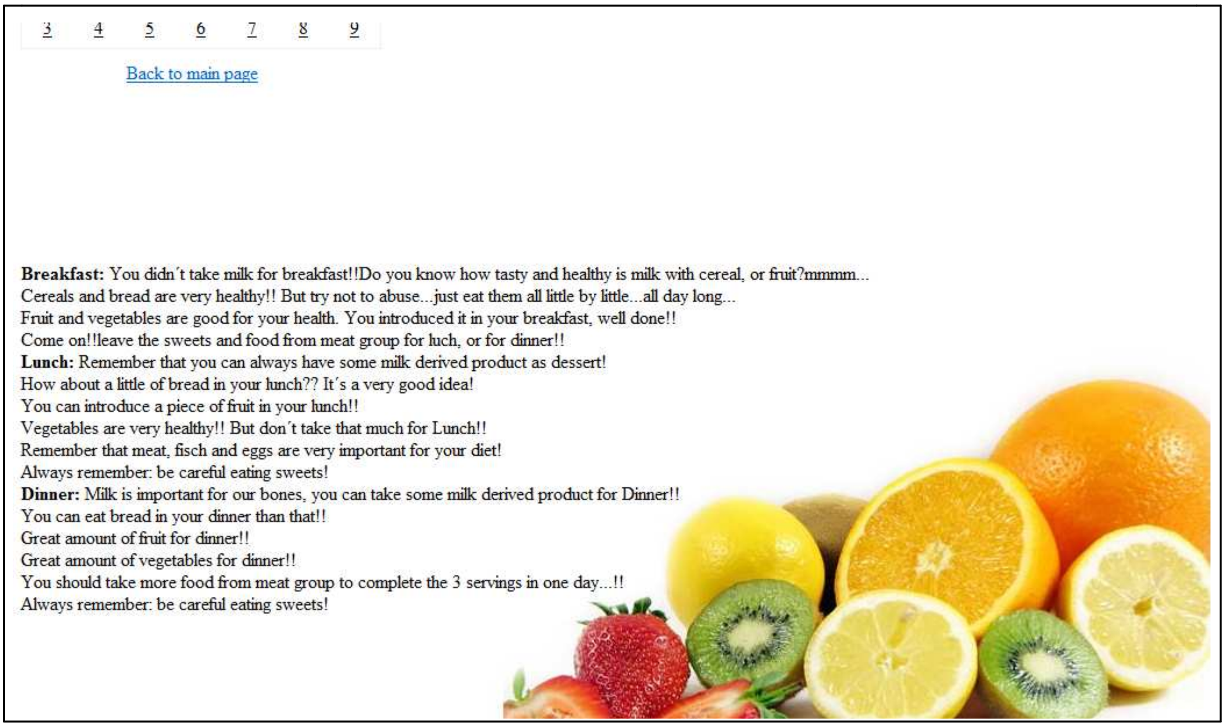
You should not eat that much sweets, fats and oils. Use them sparingly!
Milk is good for our bones!!Don´t forget it!
Proteins are important!You have to take more meat, fish or eggs!
Don´t forget a bigger amount of fruit next time!
Don´t forget bread, cereals and biscuits next time. they are good!!

<			junio de 2006			>
lun	mar	mié	jue	vie	sáb	dom
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

Back to main page

Figure 4-7: Top of the advice page with the overall advises.

The next figure shows the bottom of the advice page:



3	4	5	6	7	8	9

Back to main page

Breakfast: You didn´t take milk for breakfast!!Do you know how tasty and healthy is milk with cereal, or fruit?mmmm...
Cereals and bread are very healthy!! But try not to abuse...just eat them all little by little...all day long...
Fruit and vegetables are good for your health. You introduced it in your breakfast, well done!!
Come on!!leave the sweets and food from meat group for luch, or for dinner!!
Lunch: Remember that you can always have some milk derived product as dessert!
How about a little of bread in your lunch?? It´s a very good idea!
You can introduce a piece of fruit in your lunch!!
Vegetables are very healthy!! But don´t take that much for Lunch!!
Remember that meat, fisch and eggs are very important for your diet!
Always remember: be careful eating sweets!
Dinner: Milk is important for our bones, you can take some milk derived product for Dinner!!
You can eat bread in your dinner than that!!
Great amount of fruit for dinner!!
Great amount of vegetables for dinner!!
You should take more food from meat group to complete the 3 servings in one day...!!
Always remember: be careful eating sweets!

Figure 4-8: Bottom of the page with the specific advices.

4.5 Code behind

To carry out all the functions of the application, there are many files that work “behind the scenes” and that make it possible to change the interfaces and to make easier for the user to interact with the application.

In this section, we are going to explain the most important steps made in the development of the code, the problems faced and their solution and all the different challenges that appeared during all the development process.

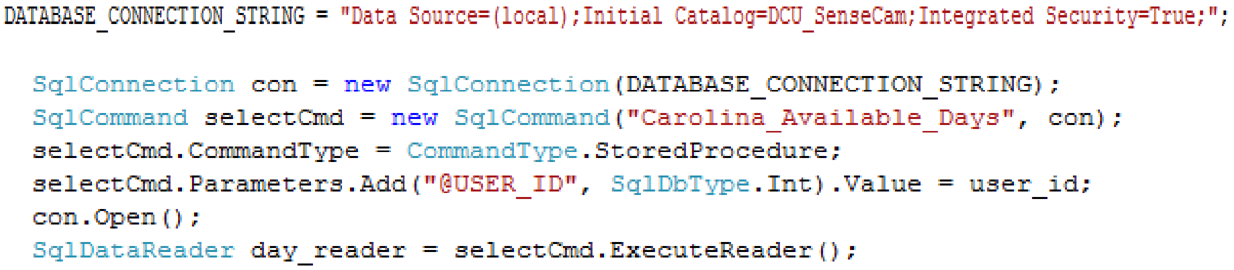
4.5.1 Main.aspx.cs

The first challenge was to be able to present a calendar initialized in the month when the last picture was taken and with the exact day marked. The second challenge was to show all the images kept in the database. The images shown are all those with an eating probability higher than 0.4. (See 4.3.1.2) Of course, there will be some days in the calendar that don´t have images, maybe because that day the user wasn´t wearing the SenseCam or because the camera didn´t take any picture where the wearer was eating. In that case, the application will inform the user that there are no images in the database for the day selected.

For that, a class called “calendar control” was made. Its methods are called from main.aspx.cs and the calendar_control class will return all the data needed for the calendar: calendar control will connect to the database.

4.5.1.1 Calendar_control.cs

From this class we will call the two stored procedures that we made before: eating images and available days. First, to load the main page, the first method in calendar_control has to return the days where we have information in: a connection to the database and the execution of “available days” stored procedure has to be made.



```
DATABASE_CONNECTION_STRING = "Data Source=(local);Initial Catalog=DCU_SenseCam;Integrated Security=True;";

  SqlConnection con = new SqlConnection(DATABASE_CONNECTION_STRING);
  SqlCommand selectCmd = new SqlCommand("Carolina_Available_Days", con);
  selectCmd.CommandType = CommandType.StoredProcedure;
  selectCmd.Parameters.Add("@USER_ID", SqlDbType.Int).Value = user_id;
  con.Open();
  SqlDataReader day_reader = selectCmd.ExecuteReader();
```

Figure 4-9: Connection with the database to execute first stored procedure

Once the calendar is initiated, it is time for the user to choose a day in the calendar. When a day is selected, we need to see all the eating images available for this day. To be able to show all images, we wrote a method which connects to the database and execute the second stored procedure: eating images. Besides showing all eating images, we will also read and save the data introduced for each of them. This is done to be able to show the user what food has he already annotated in each picture.

All data read from the database will be stored in an ArrayList, where each component is the data from one image. To help organizing this data, we wrote a class: represent image. Our ArrayList will be composed of this data type. Once all data have been retrieved and saved, we can represent it as an array of "represent_image" data type. (See Figure 4-11 for a representation of how the data is kept and extract.) Keeping everything well ordered in an array will make it easier to read and the presentation of all information will be quicker.

4.5.1.2 Represent_image.cs

The goal of this class is to be able to save the data read in an ordered way.

```
public class represent_image
{
    private int image_id;
    private string image_path;
    private string meal_type;
    private int sweets;
    private int milk;
    private int meat;
    private int vegetables;
    private int fruit;
    private int bread;

      public represent_image(int id, string path, string meal_typ, int 
sweet, int mil, int mea, int vegetable, int frui, int brea)
      {

        this.image_id = id;
        this.image_path = path;
        this.meal_type = meal_typ;
        this.sweets = sweet;
        this.milk = mil;
        this.meat = mea;
        this.vegetables = vegetable;
        this.fruit = frui;
        this.bread = brea;
      }


    public string get_path() {

       image_path= 
image_path.Replace(@"C:\nueva_carpeta\All_my_SenseCam_images\", 
"http://localhost/All_My_Images/");[2]
        return image_path; }
    public int get_id() { return image_id; }
    public string get_meal_type() { return meal_type; }
    public int get_sweets() { return sweets;}
    public int get_milk() {return milk;}
    public int get_meat(){return meat;}
    public int get_vegetables(){return vegetables;}
    public int get_fruit(){return fruit;}
    public int get_bread(){return bread;}
}
```

Figure 4-10: Class represent_image

[2] See "Problems and solutions" number **4.6.6**

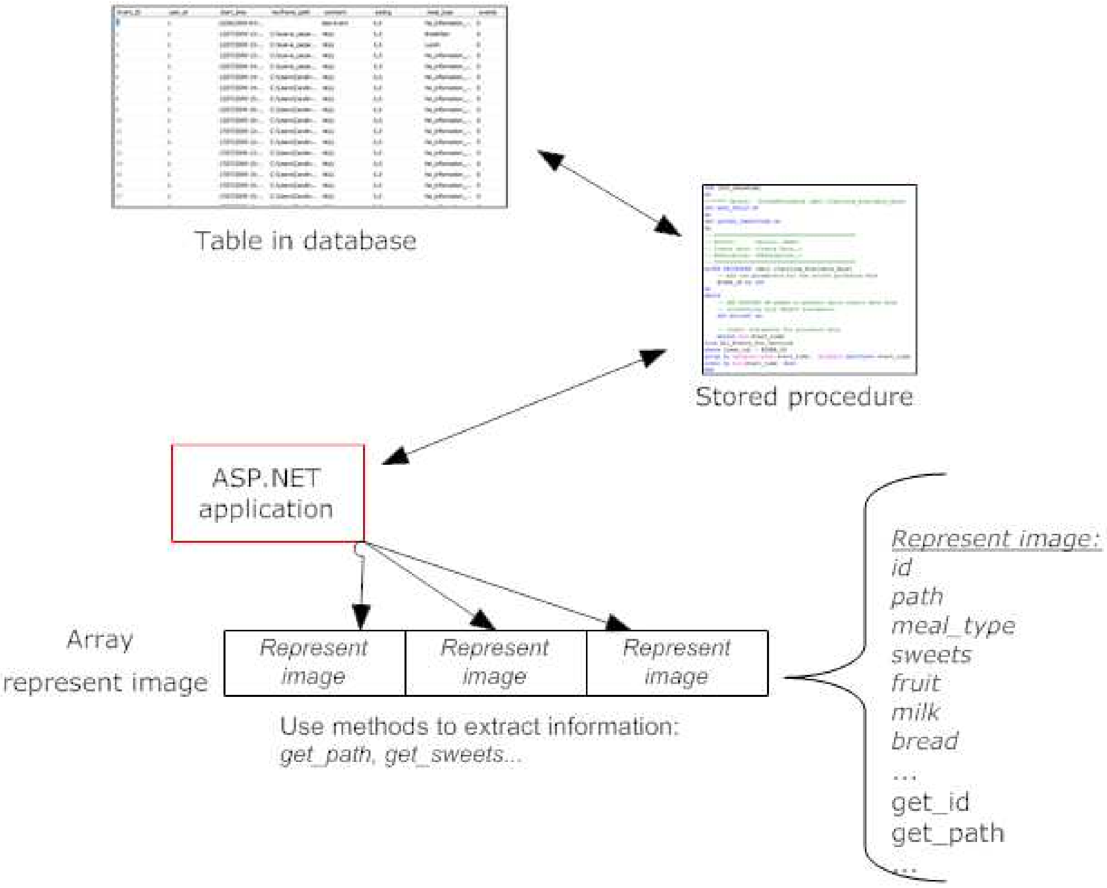
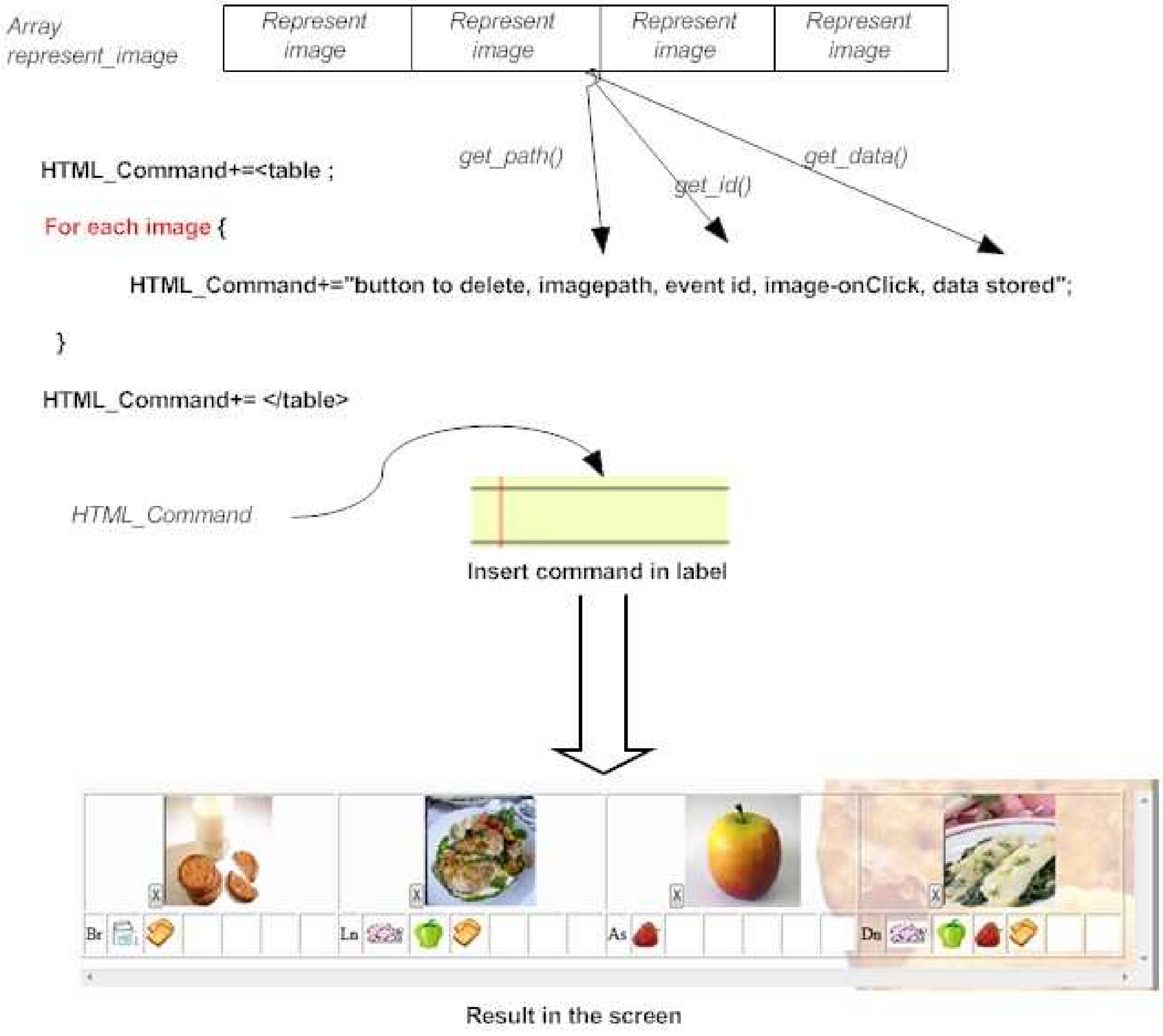


Figure 4-11: Save information in an array of data type represent_image

The images will be printed in the screen using a method called "print_images" in Main.aspx.cs. An HTML command is created to draw all the images as a table. For each image we will extract its information, kept in the array and include it in the HTML command. After creating all the HTML command, it will be represented on the screen placing it in a label. The best thing to understand this process is to take a look at figure 4-12. To see more information of how the images are displayed on the screen, see section 4.6.6.

*

To make it easier to the user to analyze the image selected and introduce all data required, the picture is shown in a bigger size. Next to it we can see some options of "meal type": breakfast, lunch, afternoon snack, dinner and extra meal. The user has to manually select one. It can be considered for future work to skip this step and automatically guess which meal of the day it is taken place revising the timestamp of the image and comparing it to the user´s profile. This different options are presented in the screen using a RadioButtonList created dynamically in the method "Add_Checklist()" from Main.aspx.cs. To select the food and the quantity taken, we tried firstly using a javascript prompt. However, it wasn´t very intuitive and the user had to type the quantities. Instead of that, we found that the best solution was to turn a food pyramid into an HTML image, where each step was a button itself. This way, the user can press the step corresponding with a group of food and automatically a slider where the number of servings can be marked appears on the screen.



```
HTML_Command+=<table ;
For each image {
        HTML_Command+="button to delete, imagepath, event id, image-onClick, data stored";
}
HTML_Command+= </table>
```

Figure 4-12: Presenting data in the screen

Once the user has introduced all the data required, all this new data has to be saved in the database in the same row as the image they belong to. For that, we will connect again with the database, but this time with a different command, one that allows us to update the table:"UPDATE table_in_database SET…".

This way, all new data will be store in the database and ready for the application to read it and use it to show and to create the advices.

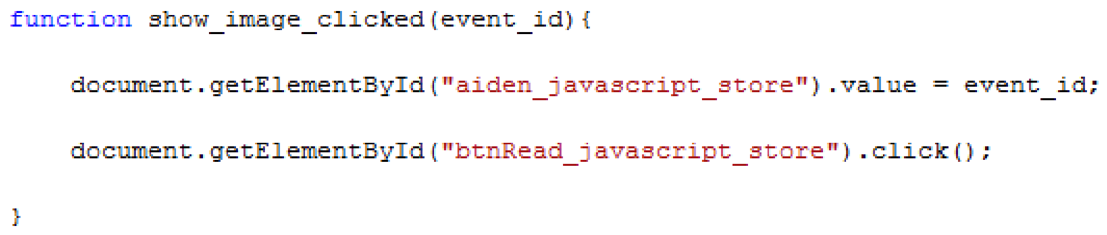
4.5.2 .aspx files

In these files we keep all what is related to the graphic interface: html and css code, but also the code needed to use Javascript. This client side language makes it easier to perform a lot of actions.

In the present project, we used it several times. All its functions are located in the main interface page: main.aspx.

4.5.2.1 show_image_clicked function

As we can see in its name, this function save the id of the image clicked by the user to be able to show it in a larger size later. When this bigger image have to be shown, we look for this variable from the aspx.cs page, identifying the image in the database and showing it again next to the RadioButtonList and the html image.



```
function show_image_clicked(event_id){

    document.getElementById("aiden_javascript_store").value = event_id;

    document.getElementById("btnRead_javascript_store").click();

}
```

Figure 4-13: Show_image_clicked function

4.5.2.2 deleteColumn function

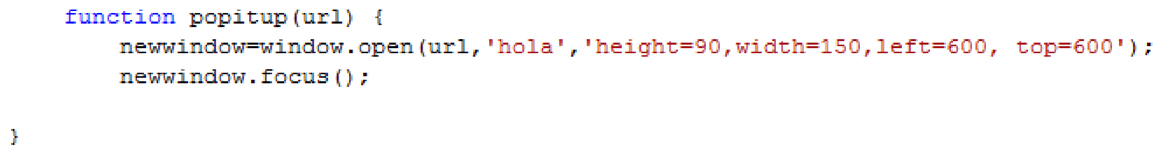
This function is used to delete a picture when it is not an eating image. When this happens, the user clicks a button besides the image (see figure 4-12). This button is "connected" with deleteColumn Javascript function, which is called this way because as the images and the information behind them are represented in a table, deleting an image is like deleting a column. Once this "X" button is clicked, a Confirm box appears asking the user if he wants to delete the image. If the answer is YES, the html code is kind of rewritten in real time and the image disappears from the screen. At the same time, its eating probability in the table of the database is changed to 0.

4.5.2.3 record_food_type_clicked function

This function is used to save all the food that the user clicks for an image in the html food pyramid. Besides, for each time called, this function itself calls another Javascript function which will open a popup that includes the slider to annotate the servings taken for the food selected. Once the user has terminated introducing new data, it can be read from the principal aspx.cs page.

4.5.2.4 Popitup function

This function is the smallest one and its only mission is to open the web form where the slider is placed.



```
     function popitup(url) {
         newwindow=window.open(url,'hola','height=90,width=150,left=600, top=600');
         newwindow.focus();

 }
```

Figure 4-14: popitup function

4.5.3 Advice creation

In the graphic beside we can see the steps follow by the application to create the advices:

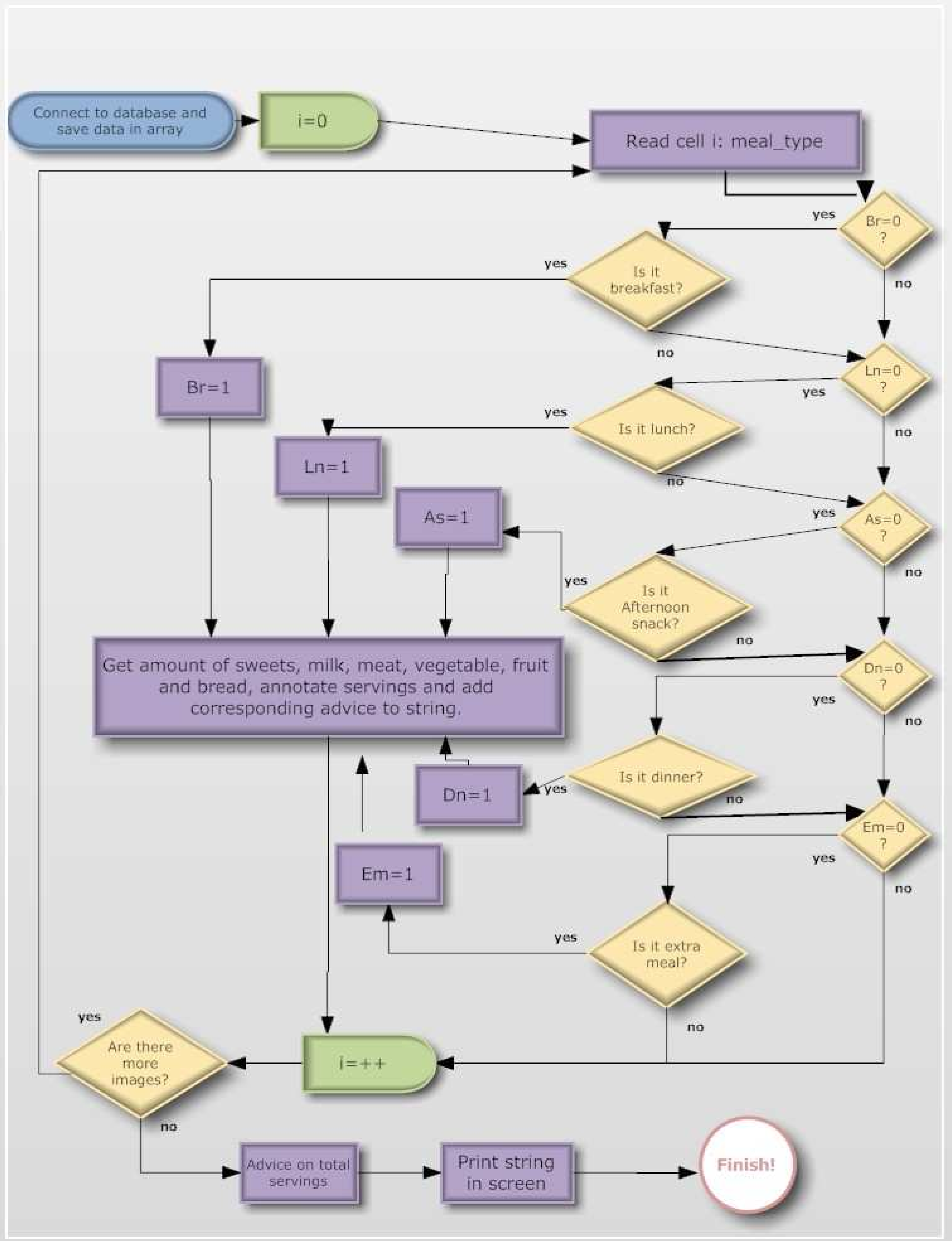


Figure 4-15: Create advice flowchart

To show personalized advices is the most important goal of this project, because it can be use for grown up to improve their eating habits and for children to educate them. The advices are created in the “Advice page”. In this page, the user selects a day in the calendar. Once a day is selected, the application reads all the information that the user has introduced into the database for that day: all the food and the quantities taken. This data read is kept in an array of “meal_data” data type, a class similar to “represent_image”. This way, it will be easy for the application to access the information and to ponder the quantities to create the required advices.

It is probable that the application shows more than one image for the same meal and we shouldn´t count the same amount of food two times. That is why we only count the first picture of each type of meal. So if we have more than one picture of the same meal event, we just ignore all the extra pictures. We can see how this is done in figure 4-15, where a flag for each of the types of meal is used to know if we have already read information from an image in the same event.

4.6 Problems and solutions

For the development of the project, we fixed different goals. Little by little. Once a point was reached, we “travelled” to the next one, then to the next one, and so on till we completed the journey. However, during this long walk, we faced different problems that required intelligent ideas and practical solutions. Those problems were both logical and programming. Sometimes they were in sql server, sometimes somewhere in any of our files in ASP.NET. In this section we will present some of them and their resolution.

4.6.1 Location of database

The first thing that we created was a web page with a calendar in it. When one day in the calendar was selected, the images of that day appeared on screen. Initially, all these pictures were kept in a database in a DCU server. To access the images, we used the DCU server ip address. Due to this, to access the images, we needed connection to the internet and be part of the DCU LAN. This seems to be a problem when we wanted to access our own images from anywhere outside DCU, which happened often.

To solve this problem, we create a table in our local database and move the information needed to our own computer. From then on, all the images and data would be downloaded to our local database. This way, we could be independent from a remote database and access our information from everywhere.

```
public static string DATABASE_CONNECTION_STRING =
"Data Source=136.206.19.54 ;Initial Catalog=DCU_SenseCam; User=Carolina;
Password=Madrid”

public static string DATABASE_CONNECTION_STRING = "Data
Source=(local);Initial Catalog=DCU_SenseCam;Integrated Security=True;";
```

Figure 4-16: Old vs. new Database Connection String

4.6.2 Day without images

Not all days in the calendar have images. At the beginning, this was a problem, because when a day without images was selected, the application returned an error.

To repair this we made “calendar control” to return a null value if the day selected had no eating images. From the principal web form we can read the null value and write a response in the screen informing the user that there are no images available for the day selected.

4.6.3 Too many images for screen size

Sometimes, there are days in the calendar that have a big amount of images. It was very difficult to see all of them in the screen and sometimes their presentation wasn´t organized enough. That is why we decided to make them appear in a line of four images and if the number of images is more than four, we can just use a scroll bar to visualize all of them. To do this, in the .aspx page, where we place the html code, we added a different “style” in the label where the images are placed:

```
style="overflow:scroll;
```

This way, when the size of images exceed the size of the label, a scroll will appear to be able to see the overflow images.

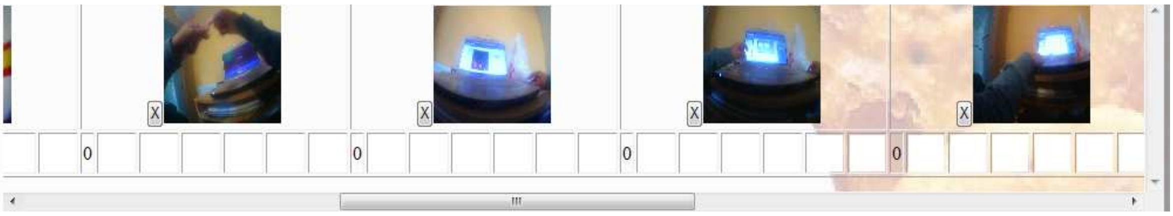
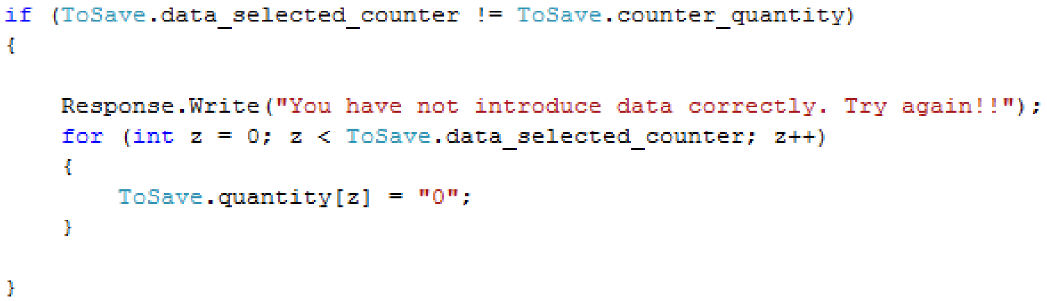


Figure 4-17: SenseCam images with scroll bar

4.6.4 Error introducing data

While the user is introducing the food that he took one particular day, some mistakes can be made. For example, he can select a type of food and forget afterwards to introduce the quantity taken. When this happened, the software became unstable and a fatal error took place. To avoid this type of errors, the program checks that the data has been introduced correctly before making further steps.

To do so, we compare the arrays where the type of food and the different quantities are stored. If there are more quantities than types of food, the program will reset the quantities already introduced and will inform the user about the mistake, asking to insert the data again. In the following figure we can see how the program performs this checking.



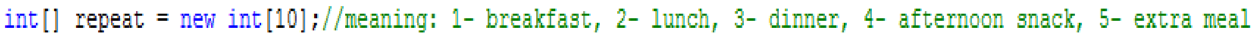
```
if (ToSave.data_selected_counter != ToSave.counter_quantity)
{

    Response.Write("You have not introduce data correctly. Try again!!");
    for (int z = 0; z < ToSave.data_selected_counter; z++)
    {
        ToSave.quantity[z] = "0";
    }

}
```

Figure 4-18: Checking errors when introducing data

4.6.5 More than one picture of the same meal

Almost one hundred per cent of the times we will have more than one picture of the same meal. This is obvious, because the SenseCam takes pictures very often and a lunch time for example, lasts between half and one hour.

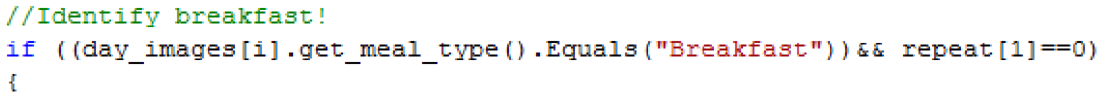
Therefore, we will have more than one picture with the same information from the same meal of a day. To avoid confusion created by repeated data or having wrong results, in the cases when two or more pictures are shown for the same meal, only data from the first picture will be taken into account.



```
int[] repeat = new int[10];//meaning: 1- breakfast, 2- lunch, 3- dinner, 4- afternoon snack, 5- extra meal
```

For this, each time that we identify a different type of meal, a flag will be placed in the position of the array that corresponds with the meal chosen.

Before saving any information about a type of meal already introduced, the program will check the array of flags:



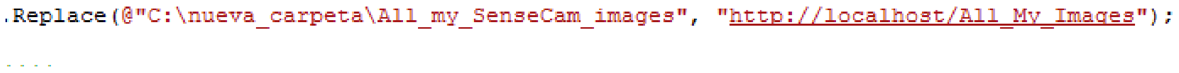
```
//Identify breakfast!
if ((day_images[i].get_meal_type().Equals("Breakfast"))&& repeat[1]==0)
{
```

Figure 4-19: More than one picture in the same meal.

4.6.6 Representation of images in the screen

To be able to show the images in the screen, it wasn´t enough to just inform the program about our picture's physical address. This way, we couldn´t present our photographs. Instead of our SenseCam pictures we had a blank space with a red cross.

To solve this problem we tried creating a virtual directory mapped to the local folder where we have all our SenseCam images. Replacing the physical address with the new virtual one, our interface was able to show the pictures normally.

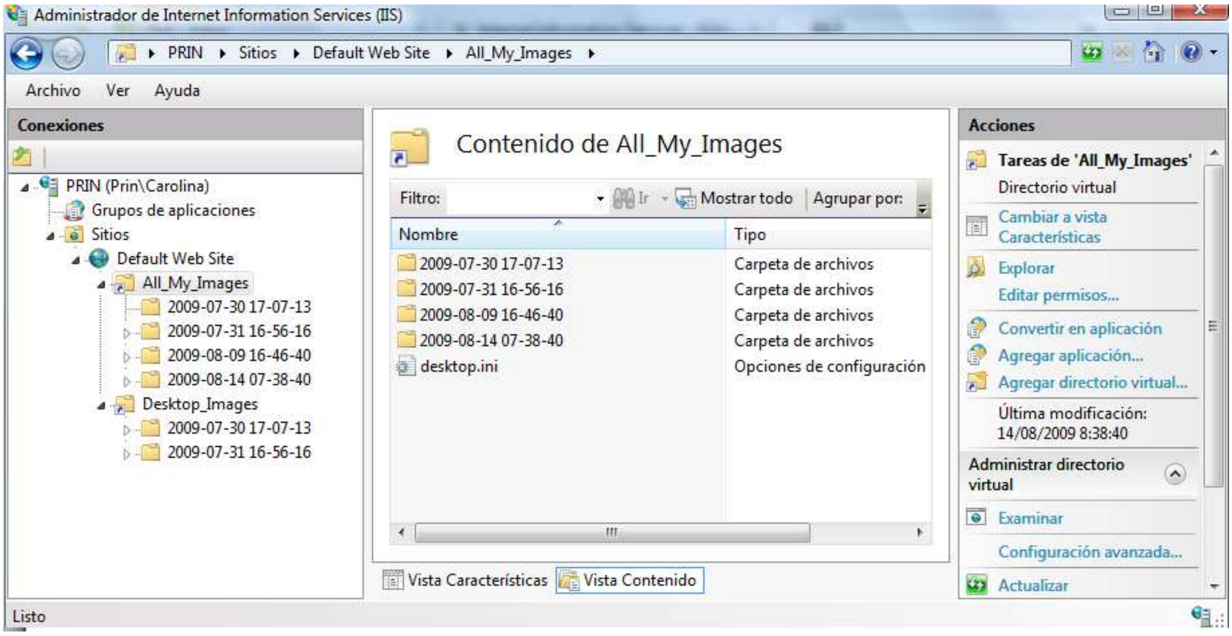


```
.Replace(@"C:\nueva_carpeta\All_my_SenseCam_images", "http://localhost/All_My_Images");
....
```

Figure 4-20: Code to replace physical address with virtual one.

To create the virtual directory, we used the Windows Vista tool IIS 7 (Internet information services).

We can see in the following figure the virtual directory created and its content in the physical folder:



Administrador de Internet Information Services (IIS)

PRIN ▸ Sitios ▸ Default Web Site ▸ All_My_Images ▸

Contenido de All_My_Images

Nombre	Tipo
2009-07-30 17-07-13	Carpeta de archivos
2009-07-31 16-56-16	Carpeta de archivos
2009-08-09 16-46-40	Carpeta de archivos
2009-08-14 07-38-40	Carpeta de archivos
desktop.ini	Opciones de configuración

Última modificación: 14/08/2009 8:38:40

Figure 4-21: Virtual directory for “All_My_Images”

4.6.7 Servings

In the first version of our software, the servings were presented in the slide as quantities from 0 to 100. This was ok for testing results, but it wasn´t easy for a normal user to count servings with accuracy. That´s why we changed the values of the slide (Always thanks to Aiden good advices). It now goes from zero to five.

5 Tests and results

In this section we will show with practical examples how useful SenseCams and this interactive software are to analyze dietry habits. We will analyze if it guides us correctly to improve our diet and if it is more or less useful than the traditional diet diary.

Besides, we will prove the precision that concept detection techniques have and if this precision is acceptable in practice.

For this, I have been wearing a SenseCam for one week and at the same time, I have been writing a traditional diet diary. To have an objective point of view, every time I had to eat, I asked someone to annotate exactly what food I was taken and what the exact quantities were. Here we have the results.

5.1 Concept detection

As we can see in table 2-7, the system misses just the three per cent of the eating images. However, between all the "positives" pictures, we have more or less half of them that are not real positives. That is why we have a very easy procedure to delete all the false positives.

In the future, the quality of the pictures taken by a SenseCam will surely improve, as well as the recognition techniques. We will have more accurate patterns and the detection will be more and more precise. I am sure that in a not very far future we will have recognition techniques that not only will tell us if a picture is about eating. They will tell us even the type of food that we are taken in a certain picture.

The most usual "failure" that the recognition system has is to confuse images from supermarkets or restaurants with eating images. This is due to the diversity of colors that both types of images have and because in most supermarket pictures the "main character" is food.

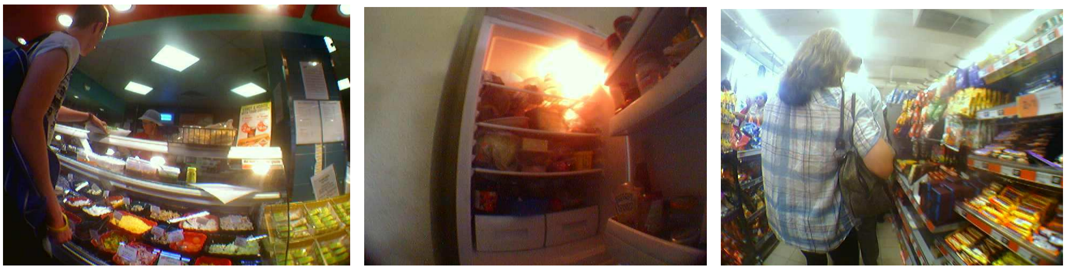


Figure 5-1: three false positives eating pictures

5.2 My week with a SenseCam

As I said before, to test how helpful is our software, I have been wearing a SenseCam for one week, and at the same time, I have been using a food diary.

To do this test, I have followed some rules:

- Wear the SenseCam all the time during the day. This way every meal was registered and we can test how well the eating images are distinguished.
- Always have the diet diary within reach and write down my meals as soon as I could or as soon as I remembered.
- Don´t fetch the SenseCams images until the test week is over.
- Count on someone else to write down an objective point of view of my meals and the quantities I took.

In the following figure we can see three pictures of the diet diary, taken with my SenseCam:

Figure 5-2: My diet diary

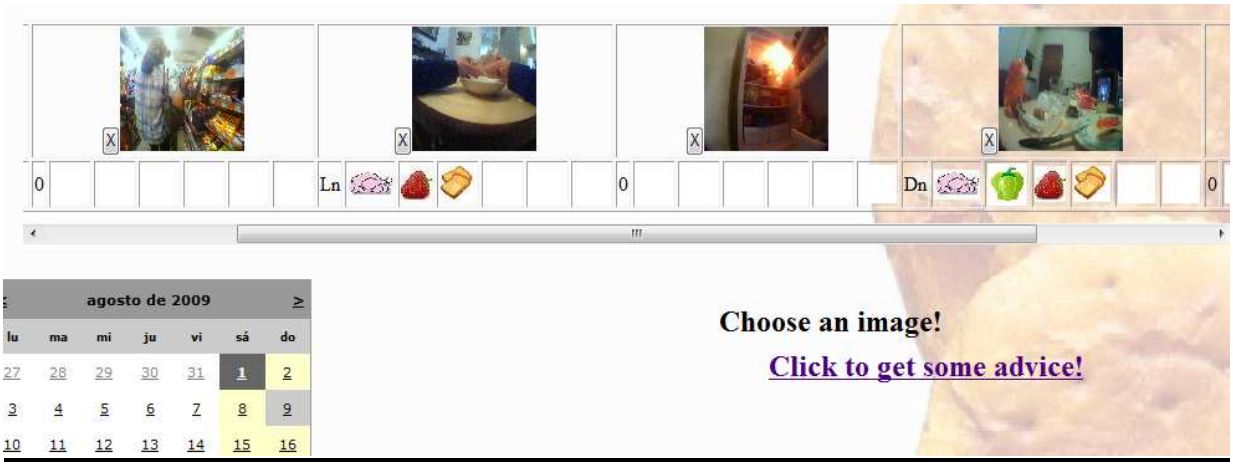
To present the results, we are going to show, with detail, three of the seven days that the test lasted. We will include the information extracted from the other four days of test in graphics and statistics that will help us to have a general understanding of the results presented.

In some of the days we happen to have a big amount of pictures, so only a representative part of them will be presented in this document.

These are the legends:

SC → Annotated using SenseCams. Br → Breakfast Ln → Lunch
D → Annotated using diet diary. As → Afternoon snack Dn → Dinner
R → Real quantities. Em → Extra meal

5.2.1 Day 1



agosto de 2009 >

lu	ma	mi	ju	vi	sá	do
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16

Choose an image!

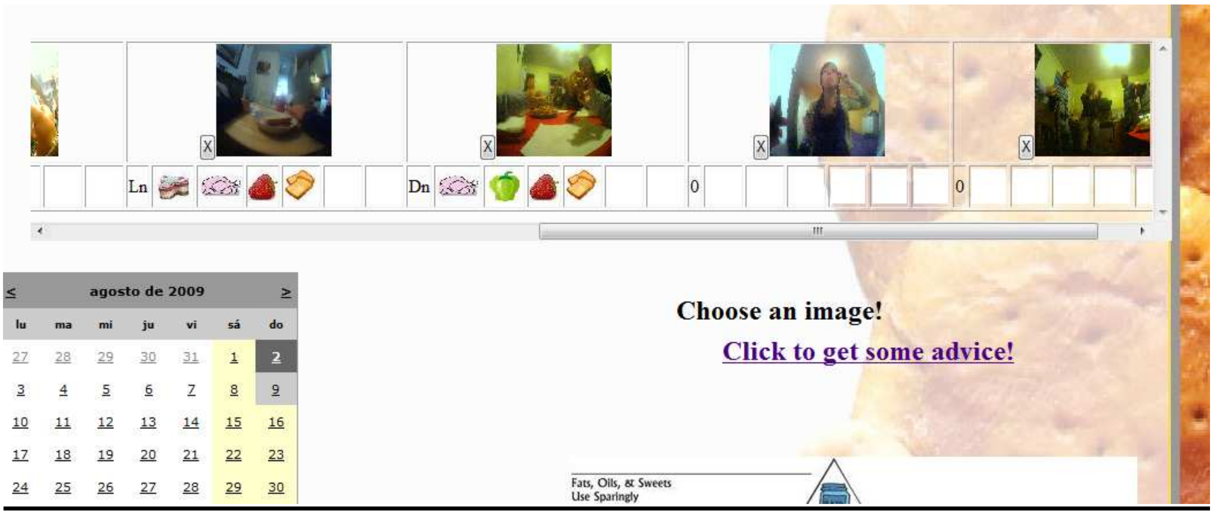
Click to get some advice!

Forgotten in diary	0 out of 3 meals				
Time written in diary	**Br**	**Ln**	**As**	**Dn**	**Em**
	Morning, day 1	Night, day 1		Night, day 1	
Nº meal´s pictures	**Br**	**Ln**	**As**	**Dn**	**Em**
	30	75		83	

(servings)	**Sweets**			**Milk**			**Meat**			**Vegetables**			**Fruit**			**Bread**		
Breakfast	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
				2	1	2										2	2	2
Lunch	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
	2	0	1										1	1	1	2	1	2
Afternoon snack	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R

Dinner	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
				1	1	1				1	1	2	1	1	1	1	1	1
Extra meal	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R

5.2.2 Day 2



< agosto de 2009 >

lu	ma	mi	ju	vi	sá	do
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

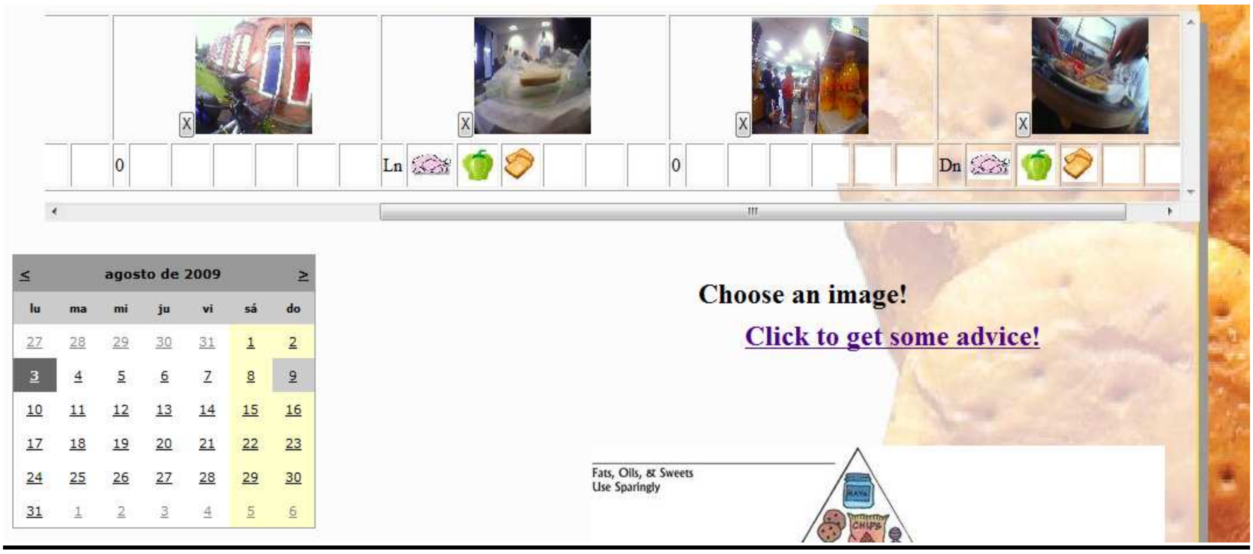
Choose an image!

Click to get some advice!

Forgotten in diary	0 out of 3 meals				
Time written in diary	**Br**	**Ln**	**As**	**Dn**	**Em**
	Noon	Noon		Night	
Nº meal´s pictures	**Br**	**Ln**	**As**	**Dn**	**Em**
	20	56		100	

(servings)	Sweets			Milk			Meat			Vegetables			Fruit			Bread		
Breakfast	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
				2	1	2										2	2	2
Lunch	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
	1	0	1				3	2	4				1	0	1	1	0	1
Afternoon snack	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
Dinner	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
							1	1	2	2	1	2	2	1	2	2	2	2
Extra meal	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R

5.2.3 Day 3



≤		agosto de 2009				≥
lu	ma	mi	ju	vi	sá	do
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

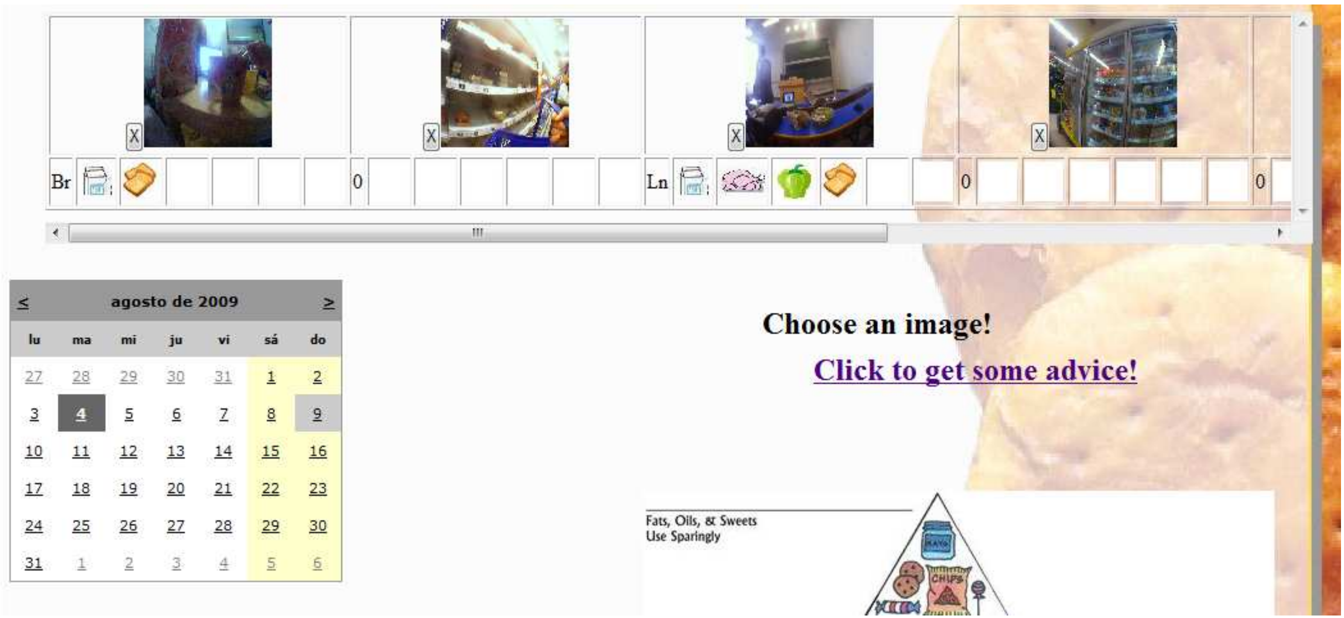
Choose an image!

Click to get some advice!

Fats, Oils, & Sweets
Use Sparingly

Forgotten in diary	1 out of 4 meals				
Time written in diary	**Br**	**Ln**	**As**	**Dn**	**Em**
	Morning	Noon		Night	
Nº meal´s pictures	**Br**	**Ln**	**As**	**Dn**	**Em**
	24	64		78	

5.2.4 Day 4



≤		agosto de 2009				≥
lu	ma	mi	ju	vi	sá	do
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

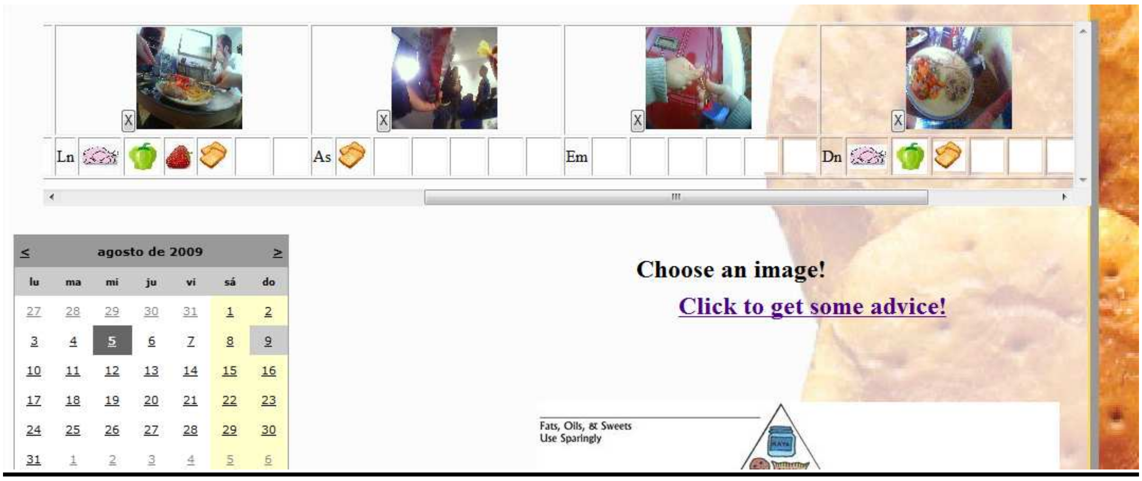
Choose an image!

Click to get some advice!

Fats, Oils, & Sweets
Use Sparingly

Forgotten in diary	0 out of 3 meals				
Time written in diary	**Br**	**Ln**	**As**	**Dn**	**Em**
	Night	Night		Night	
Nº meal´s pictures	**Br**	**Ln**	**As**	**Dn**	**Em**
	25	59		78	

5.2.5 Day 5



<			agosto de 2009			>
lu	ma	mi	ju	vi	sá	do
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

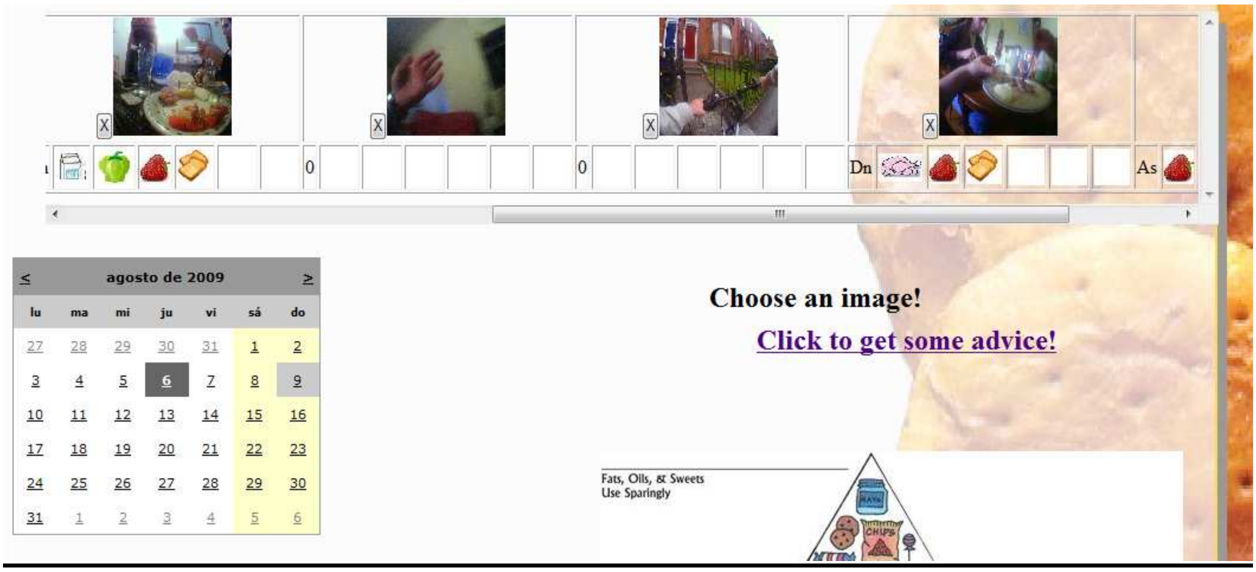
Choose an image!

Click to get some advice!

Forgotten in diary	1 out of 4 meals				
Time written in diary	**Br**	**Ln**	**As**	**Dn**	**Em**
	Noon	Noon		Night	
Nº meal´s pictures	**Br**	**Ln**	**As**	**Dn**	**Em**
	27	48	17	80	

(servings)	**Sweets**			**Milk**			**Meat**			**Vegetables**			**Fruit**			**Bread**		
Breakfast	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
				2	1	2										1	1	1
Lunch	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
							1	2	1	3	2	3	1			2	1	2
Afternoon snack	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
																2	0	2
Dinner	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R
							1	1	1	3	2	3				1	1	1
Extra meal	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R	SC	D	R

5.2.6 Day 6



≤			agosto de 2009			≥
lu	ma	mi	ju	vi	sá	do
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

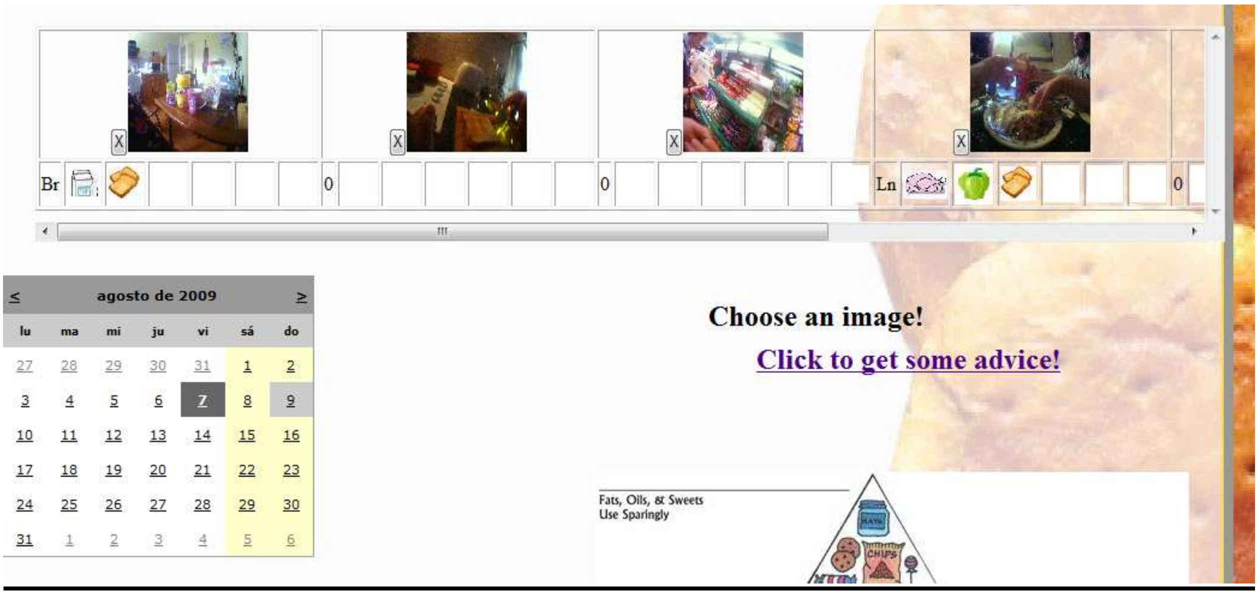
Choose an image!

Click to get some advice!

Fats, Oils, & Sweets
Use Sparingly

Forgotten in diary	**O out of 3 meals**				
Time written in diary	**Br**	**Ln**	**As**	**Dn**	**Em**
	Morning	Noon		Night	
Nº meal´s pictures	**Br**	**Ln**	**As**	**Dn**	**Em**
	20	53		66	

5.2.7 Day 7



≤			agosto de 2009			≥
lu	ma	mi	ju	vi	sá	do
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

Choose an image!

Click to get some advice!

Forgotten in diary	**0 out of 3 meals**				
Time written in diary	**Br**	**Ln**	**As**	**Dn**	**Em**
	Morning	Night		Night	
Nº meal´s pictures	**Br**	**Ln**	**As**	**Dn**	**Em**
	26	48		69	

5.2.8 Statistics

In this section we are going to present our results in a simple and clear way, showing the statistics of the most important points. This way we can compare both methods (SenseCam and traditional diet diary) and decide which one is more useful.

5.2.8.1 Annotated quantities

In the following graphic, we can see the different quantities annotated with both methods and their comparison with the real quantities. As we can see, the SenseCam method is more accurate than the diary one. This is because sometimes we wrote the meals down several hours after eating. The SenseCam method is not 100% accurate, some of the pictures are not taken from the proper angle and it can be difficult sometimes to distinguish the correct amount of food.

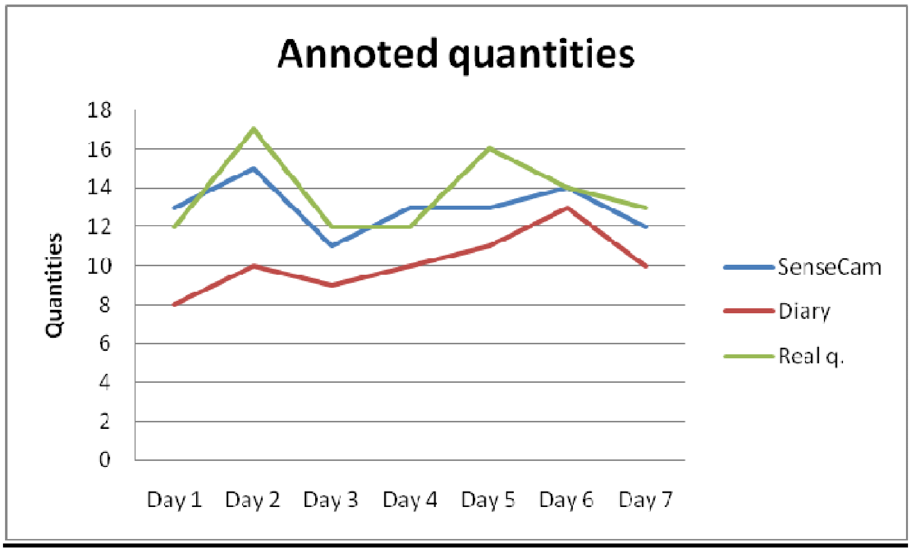


Figure 5-3: Annotated quantities

SenseCam method → 8% error.
Diet diary method → 26% error.

5.2.8.2 Number of pictures taken from each meal

The number of pictures taken in each meal depends on different factors. The most important one is the time that the meal lasts. Obviously, a longer meal will have more pictures. Besides, the frequency with which the pictures are taken can vary depending on the information collected by the SenseCam`s sensors: changes in illumination, movements, changes in temperature…

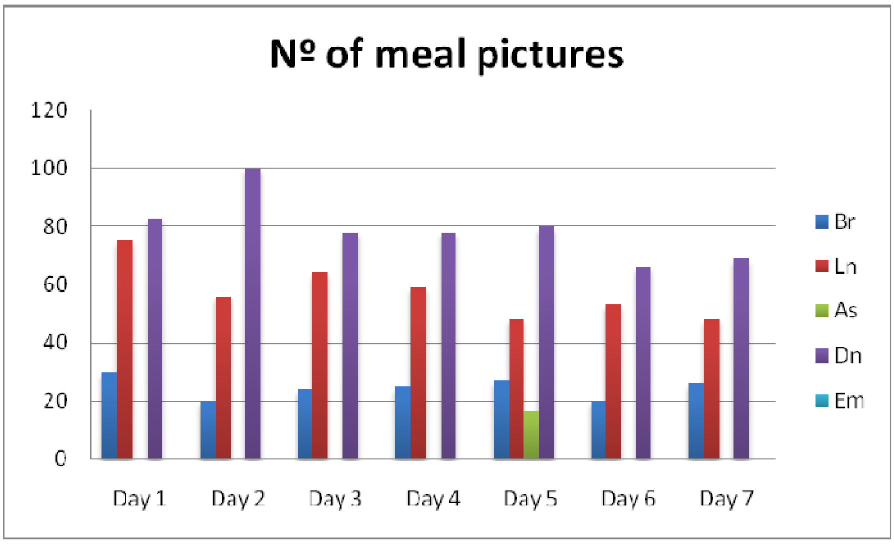


Figure 5-4: Annotated quantities

5.2.8.3 Forgotten meals

The following graphic shows the meals that we have forgotten to annotate in both methods: SenseCam and diet diary. The two meals that haven´t been written down in the diet diary were two snacks taken during the afternoon. It wasn´t a big amount of food and I forgot to write it down at the end of the day. However, in our SenseCam the pictures were recorded and it was difficult to forget about them.

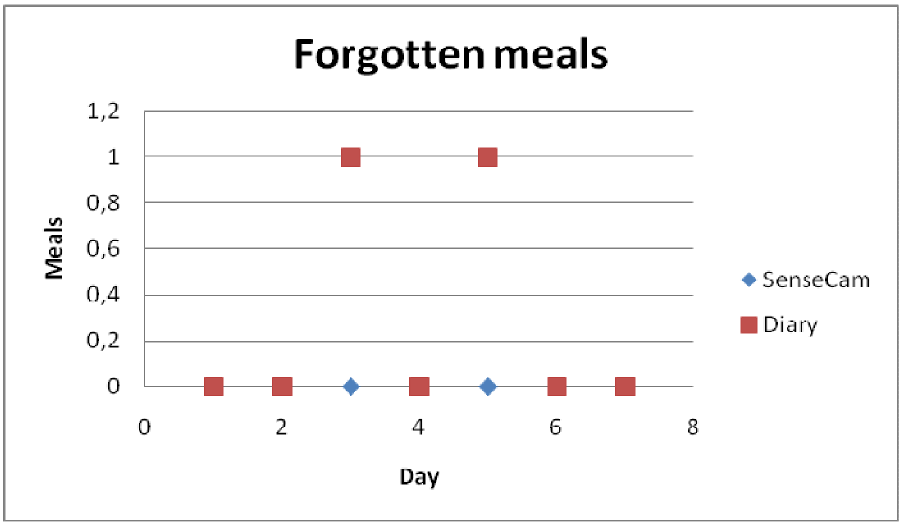


Figure 5-5: Annotated quantities

5.2.8.4 Meals annotated

In the following graphic we can see the percentage of the meals that were immediately annotated in the diet diary, and those which weren´t. Those meals that are not immediately annotated are more likely to be incomplete.

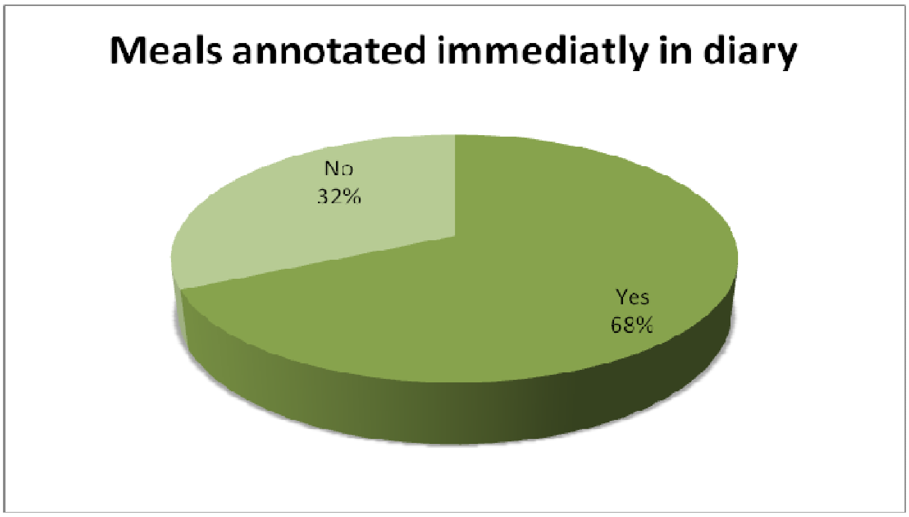
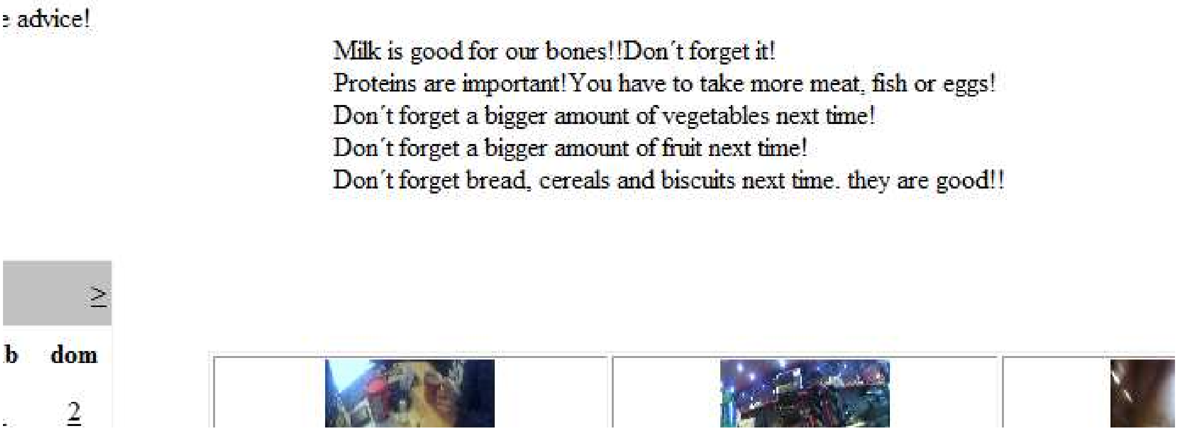


Figure 5-6: Meals annotated immediatly

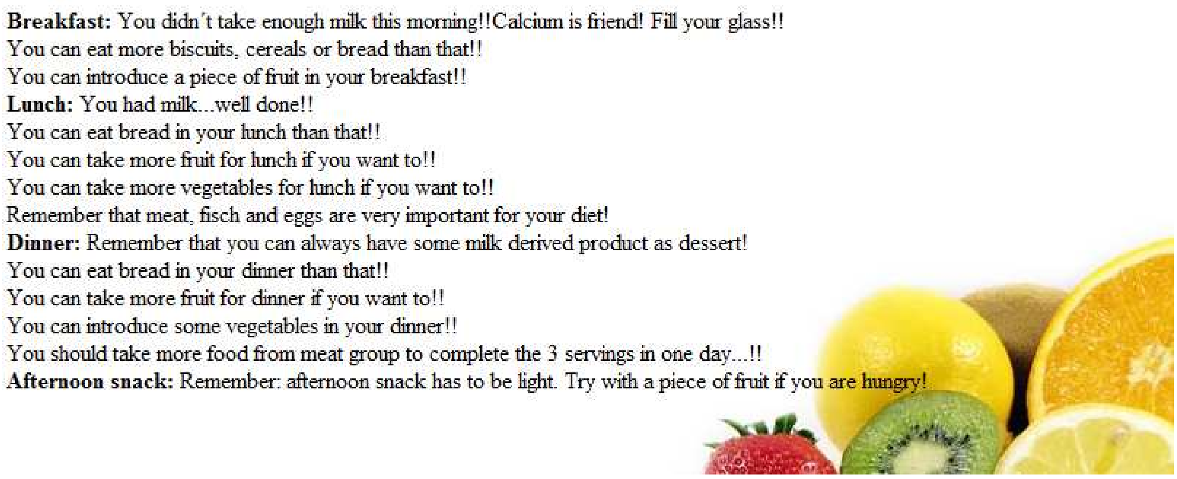
5.3 Advices

Our software creates advices based on the quantity and the type of food annotated on each picture. These are the advices I got in day 6:


advice!

Milk is good for our bones!!Don´t forget it!
Proteins are important!You have to take more meat, fish or eggs!
Don´t forget a bigger amount of vegetables next time!
Don´t forget a bigger amount of fruit next time!
Don´t forget bread, cereals and biscuits next time. they are good!!

Figure 5-7: "things to improve" advices.



Breakfast: You didn´t take enough milk this morning!!Calcium is friend! Fill your glass!!
You can eat more biscuits, cereals or bread than that!!
You can introduce a piece of fruit in your breakfast!!
Lunch: You had milk...well done!!
You can eat bread in your lunch than that!!
You can take more fruit for lunch if you want to!!
You can take more vegetables for lunch if you want to!!
Remember that meat, fisch and eggs are very important for your diet!
Dinner: Remember that you can always have some milk derived product as dessert!
You can eat bread in your dinner than that!!
You can take more fruit for dinner if you want to!!
You can introduce some vegetables in your dinner!!
You should take more food from meat group to complete the 3 servings in one day...!!
Afternoon snack: Remember: afternoon snack has to be light. Try with a piece of fruit if you are hungry!

Figure 5-8: Detailed advices.

The advices that the software gives are a guideline to have a better diet, but it´s always necessary to have a doctor following our progress and checking the steps we take. In the first part of the advice we have put together all the aspects that weren´t correct in the whole diet of the day.

In the second part, we show both correct and incorrect parts of our diet meal by meal. This way we can know specifically the things that we are doing wrong.

One thing that we can introduce in the next version of this software is the creation of not only daily but weekly and monthly advices.

6 Conclusions and future work

6.1 Conclusions

As we can see in the previous section, if we use SenseCams it is easier to remember what we took in each of our meals. Having pictures of our meals help us to remember not only the type of food we had in a certain day but how much of each nourishment we took.

It is proven that our SenseCam method is more effective and comfortable than the traditional diet notebook. Using the notebook we found out that some annotations weren't precise (26% of error), that sometimes we forgot to write down our meal (32% of the times) and that we had difficulties to remember our diet after certain time (every night before going to bed I was forced to go through my diary to check if every meal was written.) Furthermore, there are days when we have food outside the traditional hours. Most of these times I didn´t write down what I took, but everything was recorded in the SenseCam.

Regarding the creation of advices to improve our diet and therefore our health, we can see that the creation of advices with our software is instantaneous. However, if we are using the diet notebook, we will need to visit a nutritionist or a doctor to know which steps we have to take in order to follow a correct diet. The advices created by the software were designed by nutrition professionals and can be adjusted to anyone needs. This way, we will need to go to the doctor less times than with the traditional method, because with just one visit we can adjust the program to our own needs.

Besides, the SenseCam method gives us further possibilities for the control of our diet. We can send our doctor via Internet all our eating data and he will be able to follow our diet, while we are peacefully at home.

Using the SenseCam method, I realized that my diet wasn´t as good as I thought it was. Since then I try to improve it every day.

Talking about image processing and identification of events, I found out that the precision is acceptable, however, there are sometimes pictures identified as "eating events" which are not. For that, we have designed a method to delete all images that are not needed.

Conclusiones

Como podemos ver en la sección anterior, si usamos SenseCams, es más fácil recordar lo que comimos en cada una de nuestras comidas. Tener fotografías de nuestra dieta nos ayuda a recordar no solo la comida que tomamos un día cualquiera, sino la cantidad que comidos de cada alimento. Además, hay días en los que comemos fuera de las horas tradicionales. La mayoría de estas veces, no tomé nota de lo que comí, sin embargo, todo quedaba grabado en mi SenseCam.

Está probado que el método de SenseCams es más efectivo y cómodo que el tradicional bloc de notas para apuntar nuestra dieta. Cuando usamos el cuaderno, nos dimos cuenta de que muchas anotaciones no eran lo suficientemente precisas (con un 26% de error) y algunas veces nos olvidábamos de escribir la comida (32% de las veces), lo que hacía que tuviésemos dificultades para recordar lo que habíamos tomado después de algún tiempo.

En cuanto a la creación de consejos para mejorar nuestra dieta, y por lo tanto, nuestra salud, podemos observar que la creación de consejos por parte de nuestro software es instantánea. Sin embargo, si usamos un cuaderno para apuntar nuestra dieta, necesitaríamos visitar a un nutricionista o a un médico para saber qué pasos tenemos que tomar para seguir una dieta correcta.

Los consejos creados por el software han sido diseñados por un especialista en nutrición y pueden ser ajustados a las necesidades de cualquiera. De esta manera, necesitaremos ir al doctor menos veces que con el método tradicional, porque simplemente con una visita podemos ajustar el programa a nuestras propias necesidades.

Además, el método de las SenseCam, nos brinda más posibilidades para controlar nuestra dieta. Podemos enviar nuestra información al doctor a través de internet, y podrá seguir nuestra diera mientras nosotros estamos tranquilamente en casa.

Usando el método de las SenseCam, me di cuenta de que mi dieta no era tan buena como pensaba que era. Desde entonces intento mejorarla días tras día.

En referencia al procesamiento e identificación de eventos, encontré que la precisión era más bien aceptable, sin embargo, hay algunas fotografías identificadas como “eventos comida” que no lo son. Para ello hemos diseñado un método para eliminar las imágenes sobrantes.

6.2 Future work

Nowadays, SenseCams are not very little, black devices that must be worn around the neck. With the time, its design can be developed to make them even smaller and more fashionable. For example, we could find a future SenseCam as small as a button or a SenseCam that looks like a beautiful brooch. This way, they will be even more comfortable to wear and of course, more discreet.

As we can see, the interface we have worked with is simple and easy to use. This interface can be changed to show more aspects of the diet or to give more options to the user. For example, next to the quantity slide we can place an icon representing the type of food selected that gets bigger and bigger depending on the servings annotated with the slide.

Another interesting option could be to make the pyramid change each time a food group is selected, this way we can avoid repeating meals already annotated.

Another function that we can add to our software is a calorie counter. Depending on the quantities and the type of food annotated, the software would be able to inform about the calories taken in each meal. Besides, it would be a great idea to use the accelerometer included in the SenseCam to approximately calculate the number of quantities that the user burns due to exercise.

One of the new options of our software could be to introduce different “diet profiles”. This way we could use a different profile depending on our sex, age or weight. Besides, this could be very helpful if there were more than one member of the family using our SenseCam method. Besides, we can relate the number of pictures taken in each meal with the quantity of food that we ate in that meal. The bigger the amount of food, the longest time we need to finish it and the bigger the amount of pictures taken!!

It would be also interesting to create graphics that could illustrate our evolution and the improvement of our diet based on our records´s history.

Trabajo futuro

Actualmente, las SenseCams son unos dispositivos no demasiado pequeños y negros, que tienen que ser llevados colgados del cuello. Con el tiempo, su diseño puede desarrollarse para hacer dispositivos más pequeños y a la moda. Por ejemplo, podremos encontrar una SenseCam futura tan pequeña como un botón o que parezca un bonito broche. De esta manera, serán si cabe más cómodas de llevar, y por supuesto, más discretas.

Como podemos ver, la interfaz con la que hemos trabajado es simple y fácil de usar. Esta interfaz puede cambiarse para mostrar más aspectos de la dieta o para ofrecer al usuario más opciones. Por ejemplo, al lado de la barra de cantidades, podemos colocar un icono que represente el tipo de comida seleccionada y que se haga más y más grande dependiendo de la cantidad de raciones anotadas con la barra.

Otra opción interesante puede ser crear una pirámide que cambie cada vez que un grupo de comida sea seleccionado, de esta manera podemos evitar repetir comidas que ya estén anotadas. Otra función que podemos añadir a nuestro software es un contador de calorías. Dependiendo de las cantidades y del tipo de comida anotado, el software será capaz de informar sobre las calorías consumidas en cada comida. Además, sería una muy buena idea usar el acelerómetro incluido en la SenseCam para calcular aproximadamente el número de calorías que el usuario quema debido al ejercicio.

Una de las nuevas opciones de nuestro software puede ser introducir diferentes “perfiles de dieta”. De esta manera, podremos usar un perfil diferente dependiendo del sexo, edad o peso del usuario. Además, podemos relacionar el número de fotografías sacadas en una comida con la cantidad de comida que ingerimos. ¡Cuánto más grande sea la cantidad de comida, más tiempo necesitaremos para terminarla, y por tanto, más fotos tendremos!

Además, esto puede ser muy útil si hay más de un miembro en la familia que está usando el método de las SenseCam.

Sería interesante crear gráficos que puedan ilustrar nuestra evolución y la mejora de nuestra diera basado en la historia de nuestras fotografías.

References

[1] Microsoft Research, Sensors and devices group: SenseCams http://research.microsoft.com/en-us/um/cambridge/projects/sensecam/

[2] Ken Wood , Rowanne Fleck and Lyndsay Williams *"Playing with SenseCam"*

[3]Jason R. Finley, William F. Brewer, and Aaron S. Benjamin.*"SenseCam and Autobiographical Memory"* University of Illinois at Urbana-Champaign. Psychonomics 2008.

[4] ADXL210E datasheet. Analog devices, Inc., 2002

[5] Mary Lynn Garcia, "The Design and Evaluation of Physical Protection Systems." Second edition- 2007 - 351 pages.

[6] Handbook of acoustics. By Malcolm J. Crocker. Wiley-interscience. John Wiley &Sons.

[7] "Passive infraredsensor" http://itp.nyu.edu/physcomp/sensors/Reports/PassiveInfraRedSensor

[8] Lee, Hyowon and Smeaton, Alan F. and O'Connor, Noel E. and Jones, Gareth J.F. and Blighe, Michael and Byrne, Daragh and Doherty, Aiden R. and Gurrin, Cathal (2008) "*Constructing a SenseCam visual diary as a media process".* Multimedia Systems Journal, 14 (6). pp. 341-349. ISSN 1432-1882

[9] M. Hearst and C. Plaunt. Subtopic structuring for full-length document access. In *SIGIR - The 16th Annual ACM-SIGIR Conference on Research and Development in Information Retrieval*, 1993.

[10]Laila H.Shoukry "Visualizing the MPEG-7 Color Layout Descriptor" Institute for Visualization and Interactive Systems Intelligent Systems Group,Stuttgart University

[11] Aiden R. Doherty, Alan F. Smeaton, Keansub Lee & Daniel P.W. Ellis "Multimodal Segmentation of Lifelog Data" Centre for Digital Video Proessing & Adaptive Information Cluster, Dublin City University, Ireland Lab ROSA, Columbia University, New York, USA

[12] Vinay Modi, "Color descriptors from compressed images" School of Informatics. The University of Edinburgh, UK

[13] Adis Buturovic "MPEG 7 Color Structure Descriptor for visual information retrieval project VizIR" *Institute for Software Technology and Interactive Systems Technical University Vienna*

[14] M. Sezgin and B. Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electonic Imaging*, 13(1):146–168, 2004.

[15] Aiden R. Doherty and Alan F. Smeaton “Automatically Segmenting LifeLog Data Into Events” Centre for Digital Video Processing and Adaptive Information ClusterDublin City University

[16] Mark Young, Tom Radcliffe, Paul St. john, Mark Chatterley ”Improved outcomes software” www.improvedoutcomes.com

[17] Mark Montague, Javed A. Aslam “Relevance Score Normalization for Metasearch” Department of Computer Science Dartmouth College

[18] “The AceMedia project", http://www.acemedia.org, (April 2007)

[19] Aiden R. Doherty, Alan F. Smeaton1, Keansub Lee & Daniel P.W. Ellis, “Multimodal Segmentation of Lifelog Data” Centre for Digital Video Proessing & Adaptive Information Cluster, Dublin City University, Ireland LabROSA, Columbia University, New York, USA

[20] Aiden R. Doherty, Daragh Byrne, Alan F. Smeaton, Gareth J.F. Jones and Mark Hughes, “Investigating Keyframe Selection Methods in the Novel Domain of Passively Captured Visual Lifelogs” Centre for Digital Video Processing and Adaptive Information Cluster Dublin City University, Glasnevin, Dublin 9, Ireland.

[21] SEAN MCHUGH, "Cambridge in Colour" University of Cambridge, United Kingdom

[22] Aiden R. Doherty and Alan F. Smeaton, “Combining Face Detection and Novelty to Identify Important Events in a Visual Lifelog” Centre for Digital Video Processing and Adaptive Information Cluster Dublin City University

[23] Thomas Deselaers1 and Allan Hanbury, “The Visual Concept Detection Task in ImageCLEF 2008” Aachen University, Computer Science Department, Aachen, Germany, Inst. of Computer-Aided Automation, Vienna Univ. of Technology, Austria

[24] Jurie F., Triggs, B.: Creating efficient codebooks for visual recognition. In IEEE International Conference on Computer Vision, Beijing, China, 2005, pp. 604–610.

[25] Duy-Dinh Le and Shin’ichi Satoh, “Concept Detection Using Local Binary Patterns and SVM” National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan

[26] V. Yanulevskaya, J.C. van Gemert, K. Roth, A.K. Herbold, N. Sebe, J.M. Geusebroek, “EMOTIONAL VALENCE CATEGORIZATION USING HOLISTIC IMAGE FEATURES” University of Amsterdam Informatics Institute Amsterdam, The Netherlands, University Clinic of Bonn Department of Medical Psychology Bonn, Germany

[27] Daragh Byrne1, Aiden R. Doherty, Cees G.M. Snoek, Gareth G.F. Jones, and Alan F. Smeaton, “Validating the Detection of Everyday Concepts in Visual Lifelogs” Centre for Digital Video Processing, Dublin City University, Glasnevin, Dublin 9, Ireland

[28] Jesse Liberty, Dan Hurwitz, “Programming ASP.NET” O´REILLY

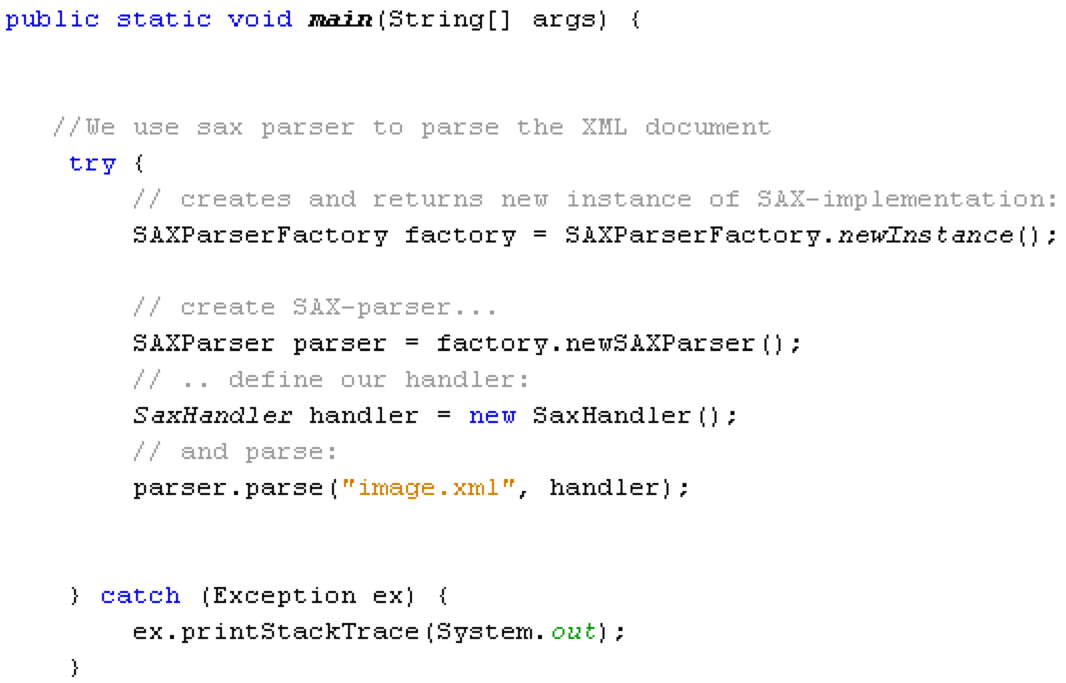
Appendix

a. Color histogram and Manhattan distance

In the first three months of this project, the main goal was to perfectly understand how the pictures` information was stored in the XML documents and how was the distinction between images and events done.

For that, we focused our attention in creating a Java program that was able to read big XML documents and able to extract all the information that we were interested in. The number of images within the same XML could be huge, so our program must be stable no matter the number of pictures stored in it. (To see how features information is stored in the XML document, see Figure 2-14)

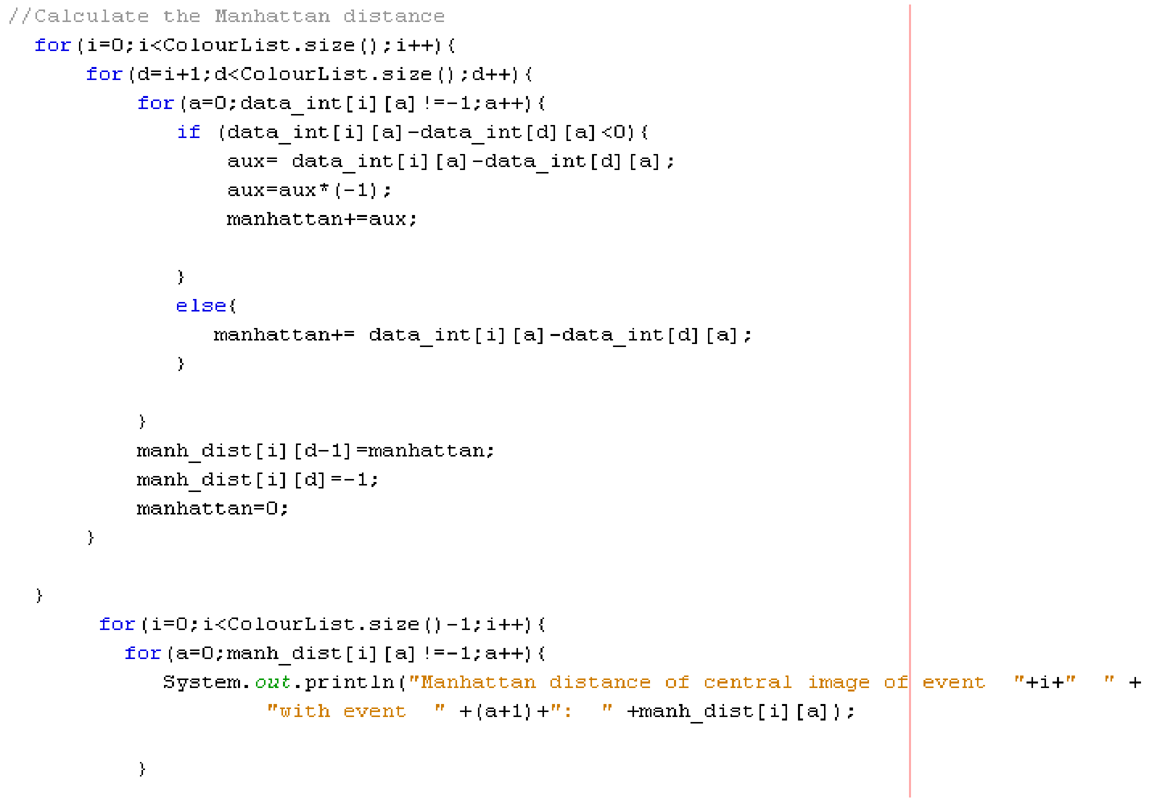
To parse the XML document we used a SAX parser. Sax parser is faster than Dom one and can handle bigger documents.



```
public static void main(String[] args) {


    //We use sax parser to parse the XML document
     try {
         // creates and returns new instance of SAX-implementation:
         SAXParserFactory factory = SAXParserFactory.newInstance();

         // create SAX-parser...
         SAXParser parser = factory.newSAXParser();
         // .. define our handler:
         SaxHandler handler = new SaxHandler();
         // and parse:
         parser.parse("image.xml", handler);


     } catch (Exception ex) {
         ex.printStackTrace(System.out);
     }
```

Figure A-1: Implementation of Sax parser in Java

To compare the different pictures we chose color histogram feature of the centre image of each of the events and made the comparison using the Manhattan distance. The reason to choose Manhattan distance as the vector distance method is that Manhattan distance is very easy to implement and the results are more than acceptable in comparison with other vector distance methods. (See Table 2-2). In the next figure we can see the algorithm used to calculate the Manhattan distance between the centre image of each of the events with the center image of the rest of them.

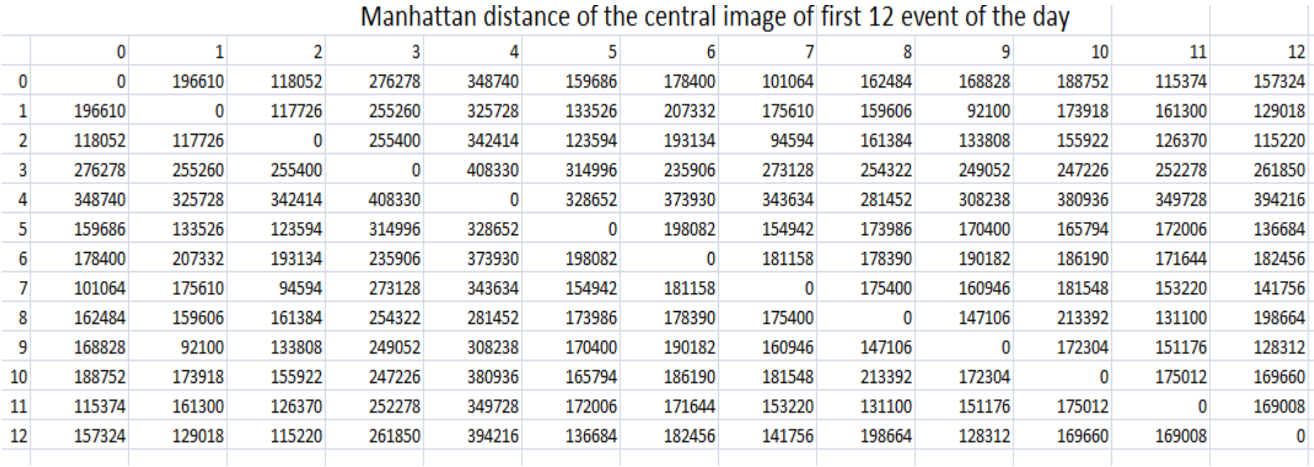


```
//Calculate the Manhattan distance
  for(i=0;i<ColourList.size();i++){
      for(d=i+1;d<ColourList.size();d++){
          for(a=0;data_int[i][a]!=-1;a++){
             if (data_int[i][a]-data_int[d][a]<0){
                 aux= data_int[i][a]-data_int[d][a];
                 aux=aux*(-1);
                 manhattan+=aux;

             }
             else{
                manhattan+= data_int[i][a]-data_int[d][a];
             }

          }
          manh_dist[i][d-1]=manhattan;
          manh_dist[i][d]=-1;
          manhattan=0;
      }

  }
       for(i=0;i<ColourList.size()-1;i++){
         for(a=0;manh_dist[i][a]!=-1;a++){
            System.out.println("Manhattan distance of central image of event  "+i+"  " +
                    "with event  " +(a+1)+":  " +manh_dist[i][a]);

          }
```

Figure A-2: Manhattan distance between the center picture of each event with the centre picture of the rest of events.

As we can see, the Manhattan distance is stored in a 2D array and the results printed in the screen.

To prove how this software works, we used SenseCam images from twenty-eight correlated events and a total of 4633 images. The results of the first 12 events are represented in the next table:



Manhattan distance of the central image of first 12 event of the day

	0	1	2	3	4	5	6	7	8	9	10	11	12
0	0	196610	118052	276278	348740	159686	178400	101064	162484	168828	188752	115374	157324
1	196610	0	117726	255260	325728	133526	207332	175610	159606	92100	173918	161300	129018
2	118052	117726	0	255400	342414	123594	193134	94594	161384	133808	155922	126370	115220
3	276278	255260	255400	0	408330	314996	235906	273128	254322	249052	247226	252278	261850
4	348740	325728	342414	408330	0	328652	373930	343634	281452	308238	380936	349728	394216
5	159686	133526	123594	314996	328652	0	198082	154942	173986	170400	165794	172006	136684
6	178400	207332	193134	235906	373930	198082	0	181158	178390	190182	186190	171644	182456
7	101064	175610	94594	273128	343634	154942	181158	0	175400	160946	181548	153220	141756
8	162484	159606	161384	254322	281452	173986	178390	175400	0	147106	213392	131100	198664
9	168828	92100	133808	249052	308238	170400	190182	160946	147106	0	172304	151176	128312
10	188752	173918	155922	247226	380936	165794	186190	181548	213392	172304	0	175012	169660
11	115374	161300	126370	252278	349728	172006	171644	153220	131100	151176	175012	0	169008
12	157324	129018	115220	261850	394216	136684	182456	141756	198664	128312	169660	169008	0

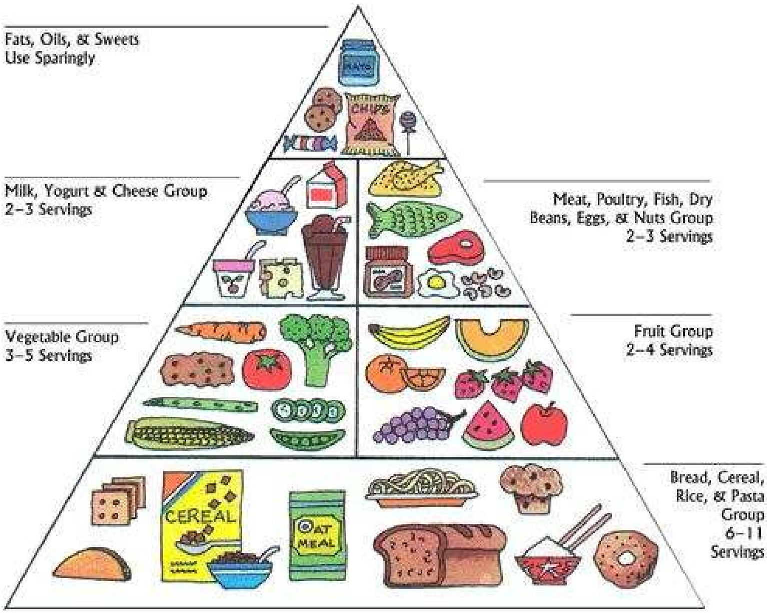
Table A-1: Manhattan distance of the central image of first 12 events of a day.

These images are represented in figure 2-20, and analyzing the graphic, we can see that those images that are more similar to events 0 and 8 have smaller Manhattan distance between their color histograms.

b. How much is a serving?

To annotate the quantity of food taken in each meal, we have to create guidelines of how much a serving is. Of course, servings will be different depending on which part of the pyramid the food belongs to.

To have a complete and varied diet, we will have to complete all the serving amounts a day that the pyramid shows:



In this application this is what we consider as a serving for each the pyramid´s groups:

Group 1, sweets: We don´t count serves from this group. The only advice is to use them sparingly.

Group 2, milk: One serving equals 1 cup of milk or yogurt, 60 grams of cheese or 2 cups of ice cream. If it is low fat, it will be better.

Group 3, Meat, fish, eggs…: One serving equals 30 grams of cooked lean meat, poultry, or fish; 1/4 cup dried beans, after cooking; 1 egg; 1 tablespoon of peanut butter; or 15 grams of nuts or seeds.

Group 4, vegetables: One serving equals 1/2 cup of raw or cooked vegetables or vegetable juice or 1 cup of leafy raw vegetables.

Group 5, fruit: One serving equals 1 cup of fruit or 100 percent fruit juice, or 1/2 cup of dried fruit.

Group 6, bread: One serving equals 1 slice of bread; 1/2 cup of cooked rice, pasta, or cereal; or 30 grams of cold cereal.

c. New food pyramid

After showing the software to our DCU nutritionist, he told us about a new food pyramid that experts in nutrition were designing.

Figure A-3: Two moments of our meeting with the nutritionist. ☺

In this new pyramid, we still have the six different groups of food, but instead of horizontal separating lines, we have vertical ones separating different colors. Each color represents one different group of food.

In this new Pyramid we make a distinction between good fat and less recommendable fat and the importance of exercise is shown. Furthermore, it teaches us that not everyone needs the same amount of food to have a correct diet.

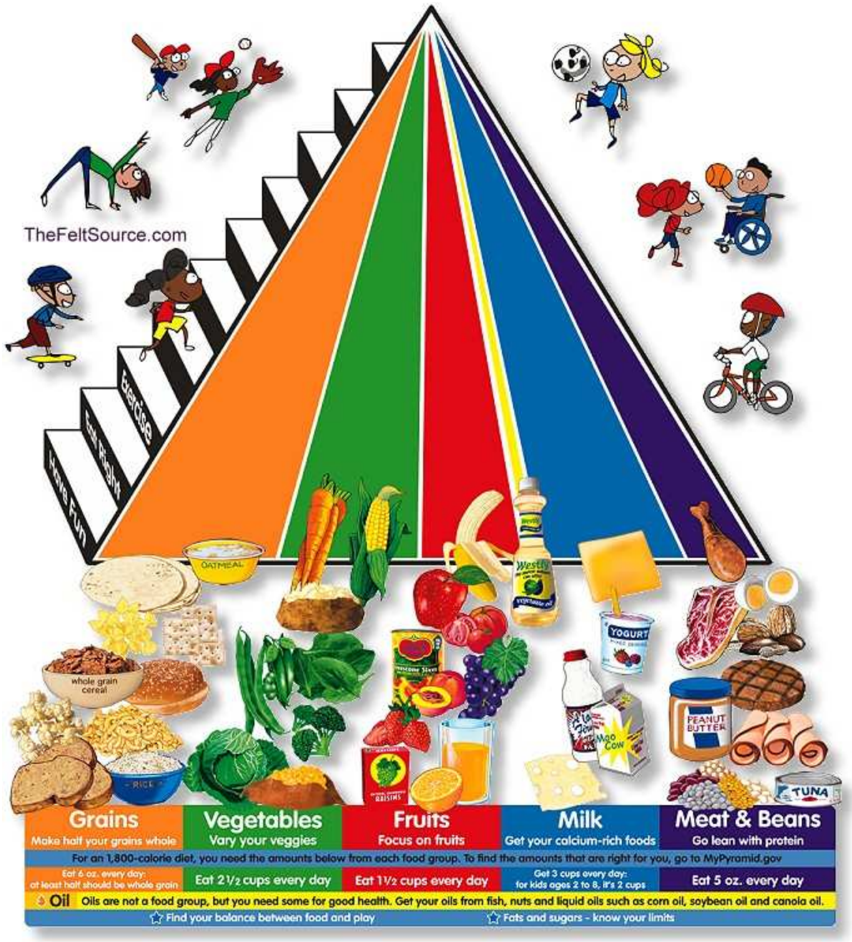

TheFeltSource.com
Exercise
Eat Right
Have Fun
OATMEAL
whole grain cereal
RICE
YOGURT
PEANUT BUTTER
TUNA
RAISINS
Moo Cow
Grains
Make half your grains whole
Vegetables
Vary your veggies
Fruits
Focus on fruits
Milk
Get your calcium-rich foods
Meat & Beans
Go lean with protein
For an 1,800-calorie diet, you need the amounts below from each food group. To find the amounts that are right for you, go to MyPyramid.gov
Eat 6 oz. every day: at least half should be whole grain
Eat 2 1/2 cups every day
Eat 1 1/2 cups every day
Get 3 cups every day: for kids ages 2 to 8, it's 2 cups
Eat 5 oz. every day
Oil
Oils are not a food group, but you need some for good health. Get your oils from fish, nuts and liquid oils such as corn oil, soybean oil and canola oil.
Find your balance between food and play
Fats and sugars - know your limits

Figure A-4: "Because one size doesn´t fit all".

PRESUPUESTO

1) Ejecución Material

Compra de ordenador personal (Software incluido)	2.000 €
SenseCam	560 €
Material de oficina	150 €
Total de ejecución material	2.710 €

2) Gastos generales

16 % sobre Ejecución Material	434 €

3) Beneficio Industrial

6 % sobre Ejecución Material	163 €

4) Honorarios Proyecto

900 horas a 15 € / hora	13500 €

5) Material fungible

Gastos de impresión	60 €
Encuadernación	200 €

6) Subtotal del presupuesto

Subtotal Presupuesto	17067 €

7) I.V.A. aplicable

16% Subtotal Presupuesto	2730.7 €

8) Total presupuesto

Total Presupuesto	19797.7 €

Madrid, Mes Marzo de 2010

El Ingeniero Jefe de Proyecto

Fdo.: Carolina Camacho García
Ingeniero Superior de Telecomunicación

PLIEGO DE CONDICIONES

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un "SISTEMA INTERACTIVO PARA EL ANÁLISIS DE HÁBITOS ALIMENTICIOS". En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda

exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.