Chapter 4

Multimedia Concepts

This chapter will discuss some important multimedia concepts that are used in the explanation of the secure multimedia database models presented in chapters six to nine. This includes background information regarding images, video, and audio that is employed as part of the techniques used within the secure image database (chapters 6, 7), secure video database (chapter 8), and secure audio database (chapter 9) models respectively.

4.1 Introduction

Multimedia includes audio and visual objects which are utilized in order to improve communication and augment its presentation. Multimedia originated in the arts and education where experimentation regarding how information is expressed is an ongoing activity. Ensuring that information is presented in a manner that the intended message is correctly understood has been a concern for many social, economic, and scientific fields; multimedia helps to achieve this.

The fairly recent advance in this subject is the storage, processing, retrieval, and presentation of multimedia using digital technology. A standard personal computer can now display and manipulate images, video sequences, and audio recordings in a digital form. The quality of the digitised animations and sound effects found in a multimedia computer system often competes with those found in dedicated systems. Multimedia systems are currently being applied in a wide variety of applications, ranging from education and business to entertainment, including specialised systems for military and research purposes. Multimedia is now perceived as a breakthrough in forming the relationship between people and computers (Gibbs & Tsichritzis, 1995).

Multimedia is a broad subject that ranges from hardware to software related issues, and would require an entire textbook just to explain some of these issues in detail. This dissertation discusses the topic secure multimedia databases with regards to the three main types of digital multimedia, namely: images, video, and audio. Only important multimedia concepts used in the presentation of the secure multimedia database models will be discussed in this chapter. Computer vision for example is a more advanced topic, although it will be briefly discussed under section 4.3 (“Concepts regarding video”) because it forms part of one of the models presented in chapter 8 (“Secure video databases”). For a more detailed discussion regarding digital multimedia in general, refer to Chapman & Chapman (2000). For a more detailed discussion regarding images, refer to Gibbs & Tsichritzis (1995) and Hearn & Baker (1994). For a more detailed discussion regarding digital video, refer to Sanchez & Canton (1995) and Stephens (2000).

For a more detailed discussion regarding digital audio, refer to Baert, Theunissen & Vergult (1992) and Shay (1995).

4.2 Concepts regarding images

4.2.1 Images and colour models

The most common type of graphics monitor is the raster-scan display which is based on television technology. Each screen point within this display is called a *picture element* or more commonly, a *pixel*. The ability for a raster-scan display system to store intensity information for each pixel makes it suitable to show realistic scenes with different shading and colour patterns. A *digital image* can therefore be defined as a two-dimensional array of pixels of varying colour or intensity. A number of file formats for digital images exist, such as GIF, Bitmap and JPEG; detailed information regarding the most common image file formats can be found at Kay & Levine (1992).

The screen resolution of a monitor is the current maximum number of pixels that can be displayed horizontally and vertically. For example, a monitor with an 800 x 600 screen resolution can display 800 pixels horizontally and 600 pixels vertically. The number of possible colour combinations available on a particular computer system is directly related to the amount of memory available; a larger amount of possible colour variations requires more available memory. A computer system that must be able to display one of 256 colours for each pixel requires 8 bits of memory for the colour information of one pixel ($2^8 = 256$). The number of bits required to represent a single pixel is called a colour model's *colour depth* or *colour resolution*. A monitor with an 800 x 600 screen resolution and a 24 bit colour resolution requires 800 x 600 x 24 = 11,520,000 bits (1,440,000 bytes) or about 1,406 KB of available memory to determine the colours of every pixel on a full screen display (Hearn & Baker, 1994).

4.2.2 Regions

The visible objects within an image are made up of one or more *regions*. Nilsson (1998) defines a region of an image to be a set of connected pixels satisfying the following two main properties:

1. A region is homogeneous i.e. the difference in intensity values of pixels in the same region is no more than a (usually small) specified value.

2. The union of all pixels in any two adjacent regions does not satisfy the homogeneity property.

A *region selection method* is an algorithm that can be used to choose the regions of an image. A few region selection methods are described in chapter 6 during the discussion of secure image databases. I also refer to an 'overlapping region' during the discussion of secure image and video databases. An overlapping region is one that covers a part of another region i.e. one of the regions is not entirely visible because the overlapping region is hiding a part of it. The regions that make up a person's arm can for example be concealed by the regions that make up another person standing in front of him/her. Figure 4.1 illustrates this concept. The picture on the left highlights two regions, A and B. The picture on the right displays the region A now overlapping the region B.

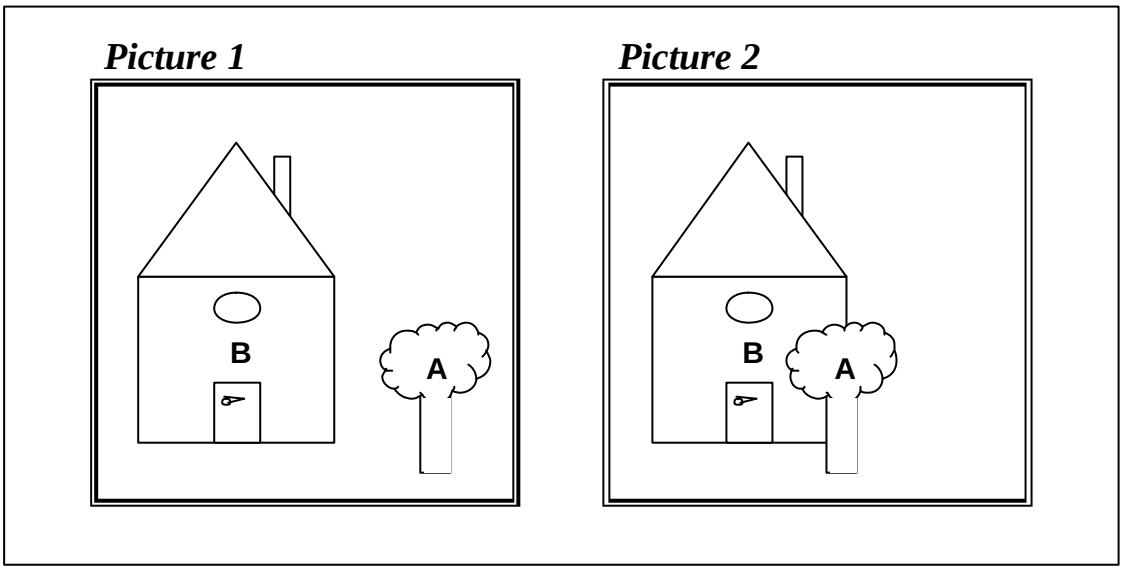

Figure 4.1: Overlapping regions

Horowitz & Pavlidis (1976) describe a region finding method called the *split-and-merge method* that can be used to separate an image into its regions. This method divides the image into four segments, and compares the adjacent segments. If the adjacent segments do not satisfy the homogeneity property, each of those segments are again divided into four smaller segments. This process is recursively repeated until every block region satisfies the homogeneity property, or until only one pixel is left in that quadrant. All the adjacent block regions (and single pixels) are compared, and are joined if their union satisfies the homogeneity property. Once all the relevant block regions and single pixels have been joined, the result will be an image that is divided into its regions.

4.3 Concepts regarding video

4.3.1 Animation and frame rate

A video file contains the information that is required to display an *animation*, which is a process in which a sequence of images, called *frames*, are rapidly displayed in order to create the illusion of motion. If the frames are shown at a fast enough pace with a small enough time difference from one frame to the next, a person would feel as though the sequence of frames is actually a single image that is changing over time. A video file often consists of a number of *scenes/cuts*, which are a group of consecutive and related frames.

At least 16 frames should be displayed per second (i.e. 16 fps) in order to ensure that the appearance of motion is smooth (when viewed by a person). To ensure this, the medium that will be used to show the video file (e.g. a computer's monitor) must be able to process and display the frames at this frame rate. Displaying the frames at a slower rate than 16 fps will most likely make the animation appear jerky. On the other hand, some of the images may never be shown if the frames are processed faster than the rate at which the display medium can update the screen. This will result in some form of *temporal aliasing*, which occurs when important visual information is removed from the animation and the outcome is an illusion of motion that does not really exist. For example, the wheels of a car that is moving from left to right could appear to be rotating anti-clockwise because some of the frames were skipped, even though the car is actually moving forward (Sanchez & Canton, 1995).

4.3.2 Simulation, scripts, and tweening

Simulation, scripts, and tweening are techniques that can be used to automatically add frames to a video file. They can also be used to add objects within a frame or a set of frames according to user input and/or a set of rules associated with the chosen technique. An animation sequence can therefore be partially or even entirely generated, saving a user from having to create each frame manually.

Simulation is a technique used to model how an object behaves over a certain period of time. This technique is used to animate an object via a program that has an associated set of rules which describe the likely or desired actions performed by the object. Simulation is commonly used to realistically automate the behaviour of an object by ensuring that these rules conform to the physical laws of the

universe. A bouncing ball may for example be realistically simulated by ensuring that the ball obeys all the laws of gravity. The simulation can be performed real-time i.e. the frames or the objects within the frames are generated as they are displayed, provided that the calculations can be performed within the required time limit (e.g. a minimum of 16 generated fps). Alternatively, the data can be simulated and only after all of the calculations are completed, the animation is visually displayed.

A *script* can also be used to generate frames or objects within a range of frames. It provides a program with the exact values regarding the positions of the objects within each frame, unlike the simulation technique which uses a set of rules to calculate the position of the objects within each frame. A script must contain all the information needed by a program to display the frames/objects correctly, i.e. the position, shape, colour, etc.

The term *key frame* originates from traditional cartoon animation. The number of frames that had to be drawn consistently often exceeded 100,000 frames (all the cartoon characters had to remain approximately the same dimension or scale throughout all these frames). To achieve this, the senior animators would draw only the frames that show a significant difference in action for each character. These *key frames* would occur often enough throughout the entire frame range to ensure that consistency among the cartoon characters is ensured once viewed at a reasonable frame rate.

Using the above approach, junior animators could then draw the frames between the key frames and constantly compare with them in order to ensure consistency. This approach would therefore dramatically reduce the total amount of time needed to draw a feature-length animation by hand. These junior animators are called *inbetweeners* or *tweeners*, and the frames drawn by these animators are called *tweens*. *Tweening* is also a technique used for digital video files in order to smoothly transform the objects in one key frame into the objects within the next key frame. It can therefore be used to automatically generate extra frames within a video file in order to make the motion appear smoother (Stephens, 2000).

4.3.3 Computer vision

The sense of vision for animals (including people) is very useful since it provides a lot of information regarding the surrounding environment, such as neighbouring objects and the estimated distance to these objects. Providing a machine with the

ability to “see” and interpret what it is “seeing” are the main concerns of a subject called *computer vision*. This is a very extensive field which consists of both general techniques and specialised techniques for certain applications. These techniques include face recognition, image interpretation, alphanumeric character recognition, and robot control.

People seem to be able to see and interpret objects without too much trouble; enabling vision for machines has however proved to be a very difficult problem. Some of the factors that make this more difficult include variable lighting, the presence of shadows for certain objects, objects that change shape such as liquids, objects that are difficult to describe, and objects that are partially covered. Computer vision has generally been more successful in man-made environments because these factors seem to be more resident in natural/outdoor scenes.

The first task to be performed in computer vision is to create one or more images of the scene. The image (if not already provided) is formed using a camera through a lens which produces a perspective or viewpoint projection of the scene. A lot of effort has gone into providing vision for mechanical robots (this subject is called *robot vision*). The visual processing required for robot vision (to interpret the objects in an image) can be simplified into two stages: *image processing* and *scene analysis*. The image processing stage involves preparing the original image for the scene analysis stage. The scene analysis stage then attempts to extract the features or objects from the processed image which will be used by the robot or agent as needed.

The image processing stage helps reduce noise, accentuate edges, and find the regions within the original image that was created from a particular scene. An *averaging operation* can be performed on the image which helps smoothen out certain irregularities within the two dimensional array of pixels (or light intensities). This involves sliding an *averaging window* over the array of pixels, and for each pixel, replacing its value with the weighted sum of all the pixel values within the averaging window. Several methods exist which can be used to extract the edges within an image (an edge is any boundary between parts of an image with visibly different values of some property e.g. intensity). The edges are then used to convert an image into a line drawing, which can help identify objects (in the scene analysis stage). A number of region finding techniques can also be used, such as the split-and-merge method discussed in section 4.2.2.

Once image processing is complete, the scene analysis stage can make use of the processed image to extract the information that is required. A large number of

images can usually be created from a particular scene. For this reason, more than one image is usually required to correctly identify objects (especially for distance related calculations), and/or more information is needed regarding the scene or objects most likely to be found in the scene. A typical technique that is used interprets a line drawing by analysing the angles made between the lines that intersect a point. This analysis can often identify concave or convex edges, and whether only one of the planes that form an edge is visible for that scene. *Model-Based Vision* is another technique which compares line drawings and regions to specified models of objects that might appear in the scene. *Stereo Vision* makes use of two or more images created from the same scene (at different viewpoints) in order to calculate the distance to objects within that scene, based on triangulation calculations (Nilsson, 1998).

4.4 Concepts regarding audio

Audio is a very different type of multimedia than the two main types that have just been discussed. Images and video files are mainly visual (if we only take into consideration the animation part of a video file). Audio, being the third main type of multimedia discussed in this dissertation, is perceived through our sense of hearing (as opposed to our sense of sight). In order to hear a sound, our ears detect the vibrations that are resident in the air, which is very different from the way in which our eyes detect light. Similarly, the techniques used to ensure logical access control for audio files at the content level is very different from the way in which logical access control is ensured for images and video files at the content level (even though the model structure is similar). The following briefly explains some concepts relating to audio files that are used in the presentation of the secure audio database models.

4.4.1 Audio basics

Many devices communicate sounds using *analogue signals*, such as the majority of the world's public telephone lines. Analogue signals are formed by continuously varying voltage levels. They are most often represented by their characteristic sine wave, which shows how the voltage levels of the analogue signals change over time.

The sine wave pattern is an example of a *periodic signal*, which means that a pattern or cycle is repeated continuously. The *period* of a signal is the time

needed to complete one cycle, and the *frequency* of a signal is the number of cycles through which a signal can oscillate in one second, measured in Hertz (Hz). A signal's *amplitude* specifies the values between which the signal oscillates. The amplitude and frequency of an analogue signal vary with time to create the sounds we hear; the amplitude reflects the volume, and the frequency reflects the pitch of the sound. These two important properties will be used in one of the secure audio database models presented in section 9.2.2. The *waveform* of any sound can be displayed by plotting its amplitude against time, which can help us characterise certain types of sounds (Shay, 1995).

An analogue signal is therefore a continuously varying signal which oscillates between two values. In contrast, a computer processes and transmits data using *digital signals*. A digital signal has a constant value for a short time, and then changes to a different value. An analogue signal may therefore at times need be converted to an equivalent digital signal (and vice versa). Figure 4.2 illustrates an example of an analogue to digital conversion. See Shay (1995) for the details of the techniques used to perform these conversions. Note that the secure audio database models discussed in chapter 9 refer to digital audio files (i.e. these files store information in digital signal form). The above concepts do however apply to both types of signals.

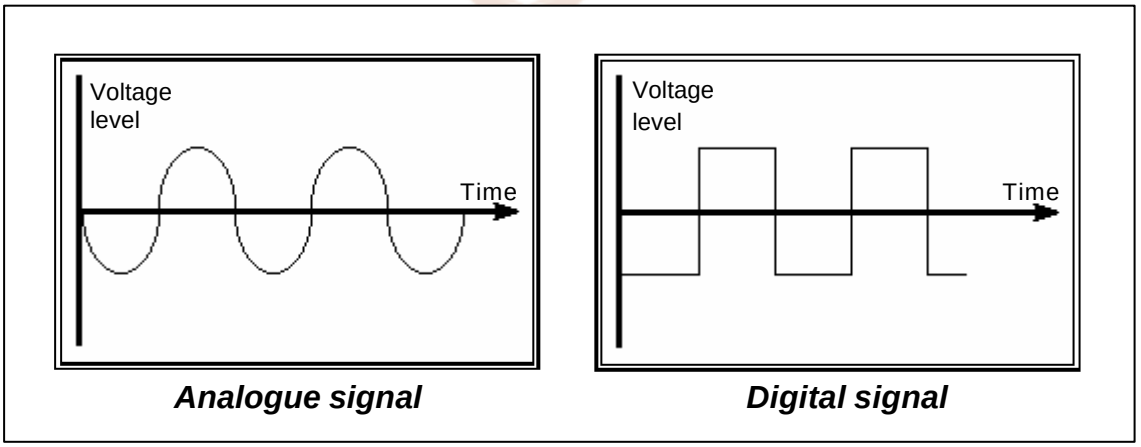

Figure 4.2: Analogue vs. Digital signal (Adapted from Shay, 1995)

4.4.2 Audio effects

Many audio effects exist which can be used for a number of reasons. They are often used to change the quality of a sound, ranging from minor improvements to major modifications that compensate for poor performance and recording. Some audio effects are even used to dramatically change particular sounds (whether to

mask the contents or for entertainment purposes), while other effects create new sounds out of the original. An audio effect may be applied to an entire audio file, or only to a specified time range within the original audio file. Although traditional techniques such as *de-essers* (used to remove sibilance after talking or singing too close to a microphone) and *click repairers* (used to remove clicks from damaged vinyl records) can transform analogue signals to improve quality, most audio effects are specifically designed to be used with digital sound, allowing a user to modify the sound (usually using a computer) as required.

Graphic equalisation is often used to transform the spectrum of a sound using a bank of filters which can for example produce a desired frequency balance or artificially enhance the bass. *Envelope shaping* operations can change the outline of a waveform, thereby modifying the amplitude as needed e.g. for fade-in and fade-out effects. *Time stretching* is used to change the duration of digital sound without changing the pitch, often used to synchronise audio with video or another sound. *Pitch alteration* is used to modify the pitch without affecting the duration of digital sound. Many other audio effects exist that fall into the category *'creative sound effects*'. These include effects such as reversal, flanging, mechanisation, phasing, echo, etc. (Chapman & Chapman, 2000).

4.5 Conclusion

This chapter discussed the important multimedia concepts that will be used during the explanation of the secure multimedia database models to be presented. This included key concepts surrounding images, video, and audio; the three major types of digital multimedia that are dealt with in this dissertation.