Chapter 10: Ten major achievements

Lausanne Senior Meeting

10.1. Reliability-ordered cascade

The multimodal signal community has made major progress on reliability ordered cascade for fusion of multimodal signals. Murat Tekalp’s team contributed in this domain with biometry applications in mind. Different strategies are possible: In the so-called “early integration” modalities are fused at data or feature level, whereas in “late integration” decisions or scores resulting from each unimodal classification are combined to give the final conclusion. This latter strategy is also referred to as decision or opinion fusion and is effective especially when the contributing modalities are uncorrelated and thus the resulting partial decisions are statistically independent. Early integration techniques, on the other hand, can be favoured if they are used properly and a pair of modalities is highly correlated, as in the fusion of audio and lip movement. Multimodal decision fusion can also be viewed from a broader perspective as a way of combining classifiers, which is a well-studied problem in pattern recognition. The main motivation of Murat Tekalp and his team’s work is to compensate for possible misclassification errors of a certain classifier with other available classifiers and to end up
SIMILAR Dreams

with a more reliable overall decision. Misclassification errors are in general inevitable due to numerous factors, such as environmental noise, measurement and modelling errors, or the time-varying characteristics of signals.

Recent work uses decision-level fusion for the verification problem, where scores resulting from each classifier are concatenated to form a feature vector, which is then fed into another classifier, e.g. a median radial basis function (MRBF) network or support vector machines and Bayesian classifier.

The work of Murat Tekalp and co-workers presents a new multimodality fusion strategy where some of the modalities might be corrupted by measurement noise and/or modelling errors. The basic idea is that a single highly reliable modality alone may sometimes yield a correct decision, whereas its linear fusion with some other less reliable modality may give incorrect results. On other occasions, results obtained by the fusion of two modalities may outperform those obtained from each modality alone. Hence, their proposed scheme considers all possible linearly fused modality combinations (including single modalities) with their corresponding reliability measures and strives to maximise the benefit of multimodal fusion so that the upper bound for the system error rate becomes the expected occurrence rate of the cases where all classifier combinations fail. Thus, a critical feature of their system is to be able to assess each modality classifier adaptively with a reliability measure.

10.2. Mutual information in multimodal SP

10.2.1. Mutual information

In probability theory and information theory, the mutual information of two random variables is a quantity that measures the independence of the two variables.
If $X$ and $Y$ are discrete random variables and $f(x,y)$ is the value of their joint probability distribution at $(x,y)$, and $f(x)$ and $f(y)$ are the marginal distributions of $X$ and $Y$, respectively, then:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} f(x,y) \log \frac{f(x,y)}{f(x)f(y)}$$

is the mutual information (MI) of $X$ and $Y$.

Figure 47: Mutual information formula

Informally, mutual information measures the information of $X$ that is shared by $Y$. If $X$ and $Y$ are independent, then $X$ contains no information about $Y$ and vice versa, so their mutual information is zero.

10.2.2. Mutual information in multimodal signal processing

Most of the time the various modalities that are processed consist of signals that are more or less related to each other. It was thus obvious to use the information-theoretic notion of mutual information.

This is obviously the case when several medical images of the same area acquired through different modalities are to be registered. For two images, mutual information is a measure of how well one image explains the other, or vice versa, and is assumed to be maximal if the images are geometrically aligned. Maximisation of mutual information is a very general and powerful criterion because no assumptions are made regarding the nature of this dependence and no limiting constraints are imposed on the image content of the modalities involved.
SIMILAR Dreams

Another example of the use of mutual information is the combination of audio and image or video as the voice modality will have an impact on at least one other modality, namely, the image of the face, as the person will open his/her mouth to emit sounds.

Figure 48: MI between image and audio:
The maximum clearly lies at the speaker’s mouth

10.3. Multimodal emotion detection

10.3.1. Introduction

Human communication is of course composed of verbal or explicit discourse, but it also includes many implicit components, such as emotions. Several studies depict the importance of non-verbal behaviour in human-human communication. These studies show that more than half of semantic exchanges happen through non-verbal behaviour (gestures, facial expressions, postures, etc.)

The ability to understand such emotions is then highly desirable for the computer in HCI, especially in applications where the computer has to play the role of a teacher, helper, or companion.
Emotion recognition systems will allow natural interactions between computer and users, as well as learning users’ preferences.

Information about emotions can be acquired through facial expressions, but also through language, gestures, and other physiological components (clamminess of hands, etc.)

It has been proven that humans behave purely multi-modally when recognising emotions. This area was thus an obvious choice for multimodality research.

10.3.2. Achievements

Many studies were conducted about the various physiological signs of emotions, with the face in first position. As a result of these studies, six universal face expressions were documents, namely, happiness, sadness, anger, fear, surprise, and disgust.

Other studies were conducted concerning voice, gestures, and so on.

Figure 49: Emotion recognition through the voice
SIMILAR Dreams

One robust multimodal interface has been set up for the recognition of emotions. It effects a fusion between face and prosody, i.e., the melody or musical component of the spoken voice.

10.3.3. Future research work

Research to improve robustness and to extend the scope of detected emotions is ongoing.

This work either integrates other modalities, such as gesture, or uses other fusion methods that are deemed to be more robust.

![Multimodal Fusion Diagram](image)

Figure 50: A scheme of ongoing research in emotion recognition

10.4. Multimodal biometrics

10.4.1. Introduction

Biometrics deals with authentication and identification based on a person’s characteristics. Each person has a unique anatomy and behaviour.

Applications are numerous. For example, a multi-player computer game can interact naturally with the players by recognising them. Other examples include access control to
SIMILAR Dreams

restricted secure area, prisoner monitoring, passenger identification in airports, and secure virtual private network access. Access can be controlled using a person’s face image, iris scan, and/or a few seconds of speech.

Our brains recognise and check our peers’ identities hundreds of times a day with very little error. This task is performed using visual information, for example by looking at the person’s face or gait, or auditory information, for example by listening to a person’s voice, or both.

This shows that our brains are able to extract the relevant features from the information flow that we perceive. These features are:

(i) discriminative;
(ii) compact, so as to be memorised easily; and
(iii) retrieved very quickly so as to be matched against new incoming data to recognise.

If we could determine these features and the accompanying recognition process, the identity check or recognition task could be automated, opening the doors to dozens of applications in data protection, security, and entertainment. Given two face images, we must determine automatically whether they are two images of the same person or of different persons. The field of automatic personal identity verification and recognition using human specific characteristics is called biometrics. The problem is difficult due to many factors, such as variability of facial appearance, sensitivity to noise, template aging, etc..

10.4.2. Different Fusion Strategies

We can overcome some of these difficulties by combining different information sources for the classification/ recognition task. Strategies for combining the different information sources, i.e. fusion strategies, have been proposed in order to improve the verification’s accuracy. As all algorithms operate on the same modality (face modality in this case), this fusion type is referred
SIMILAR Dreams
to as intramodal fusion. This is an example of fusion of different processing algorithms working on the same signal.

Figure 51: Intramodal fusion: One signal, several processing steps
Another fusion aspect that can be considered is the fusion of confidences obtained sequentially on several video frames of the same person's face. Instead of making the decision based on a single image, several images are taken of the person whose identity is to be verified. This type of fusion is referred to as Multi-frame fusion.

Figure 52: Multi-sensor fusion. Several cameras capture the same modality but at different angles from the subject.
The Multimodal fusion of face and speaker verification algorithms illustrates the case of different sensors capturing different physical phenomena related to the same state of nature.

In this case, the speech and face of the same subject are recorded and processed by different algorithms, which yield separate opinions. These two outputs (opinions) are then conciliated at the fusion stage.

Figure 54: Multimodal fusion Different sensors, such as a camera and a microphone, record different physical phenomena related to the same identity.
SIMILAR Dreams

Much theoretical and experimental evidence shows that the best accuracy improvement is obtained with multimodal fusion. The main reason for this is that the noise perturbing the biometric signals is much less correlated (or even independent) for two different modalities in comparison with the other fusion schemes presented here.

10.5. Multimodal medical image registration

10.5.1. Image registration

Image registration can be defined as the spatial alignment transformation (mapping) used to match two or more pictures. The registration is classified according to the origin of the differences existing between the pictures to match.

It is thus:

- multimodal if the ways of acquiring the pictures are different (e.g., various sensors),
- temporal if the time varies (e.g., before and after surgery),
- viewpoint if registration is used to infer 3D information from images in which either the camera or the objects in the scene have moved,
- template for model-based object recognition.

Medical image registration is used for three main applications. The first one, on which we shall focus hereafter, deals with different images of the same individual acquired by different means.

The other two, which entail a single modality, treat either pictures from multiples patients or images from the same patient at different times.
10.5.2. Multimodal medical image registration

The need for multimodal medical image registration comes from the rather extended number of medical imaging technologies, each of which has its own advantages:

- some primarily depict anatomy/morphology (CT, MR, etc.), whereas some primarily depict information about the metabolism of the underlying anatomy (PET, IMRI, EGG, etc.). This is the basic distinction between anatomical and functional imaging,

- within the anatomical modalities, some – for example, X-rays - are dedicated to bones, whereas others – such as US - give images of soft tissues.

Image registration lets one integrate different images into one representation such that the complementary information can be accessed more easily and accurately. Multimodal images of the same person or of different persons generally differ by local geometric differences. Consequently, non-rigid or elastic transformations are required to map such images onto one coordinate system.

The resulting aligned images give medical professionals a much better view of the imaged region. Medical image registration has been used in the diagnosis of breast cancer, cardiac studies, and a variety of neurological disorders, including brain tumours. It also allows better surgery planning and simulation as well as intra-operative navigation.

Many registration algorithms exist.

To perform the matching, several various criteria and systems (neural networks, statistical inferences, maximisation of mutual information, etc.) are used in several domains, from spatial to wavelets.
10.6. ICARE

10.6.1. Introduction: The CARE properties

The CARE properties - the Complementarity, Assignment, Redundancy, and Equivalence that may occur between the interaction techniques available in a multimodal user interface - have been designed as a simple way of characterising and assessing aspects of multimodal interaction.

Complementarity: Modalities of a set M must be used in a complementary way to reach state s' from state s within a temporal window; if all of them must be used to reach s' from s, i.e., none of them taken individually can cover the target state.
SIMILAR Dreams

To express this adequately, we need to extend the notion of reachability to encompass sets of modalities: \( \text{REACH}(s, M, s') \) means that state \( s' \) can be reached from state \( s \) using the modalities in set \( M \).

\[
\text{Complementarity} \ (s, M, s', tw) \\
\square (\text{Card}(M) > 1) \land (\text{Duration}(tw) \neq \infty) \\
\land (\forall M' \in PM (M' \land M \Rightarrow \neg \text{REACH} (s, M', s'))) \\
\land \text{REACH} (s, M, s') \land (\forall m \in M \cdot \exists t \in tw \cdot \text{Active}(m, t))
\]

Figure 57: Complementarity formula

**Assignment:** Modality \( m \) is assigned in state \( s \) to reach \( s' \) if no other modality is used to reach \( s' \) from \( s \). In contrast to equivalence, assignment expresses the absence of choice: either there is no choice at all to get from one state to another, or there is a choice but the agent always opts for the same modality to get between these two states. Thus we can define two types of assignment:

\[
\text{StrictAssignment} \ (s, m, s') \square \text{Reach} (s, m, s') \\
\land (\forall m' \in M (m' \neq m \Rightarrow \neg \text{Reach} (s, m', s'))) \\
\text{AgentAssignment} \ (s, m, M, s') \square (\text{Card}(M) > 1) \\
\land (\forall m' \in M \text{ Reach} (s, m', s') \land (\exists m \in M (m = m'))
\]

Figure 58: Assignment formula

**Redundancy:** Modalities of a set \( M \) are used redundantly to reach state \( s' \) from state \( s \), if they have the same expressive power (they are equivalent) and if all of them are used within the same temporal window, \( tw \). In other words, the agent shows repetitive behaviour without increasing its expressive power:

\[
\text{Redundancy} \ (s, M, s', tw) \square \text{Equivalence} (s, M, s') \\
\land (\forall m \in M \cdot \exists t \in tw \cdot \text{Active}(m, t))
\]

Figure 59: Redundancy formula
**SIMILAR Dreams**

**Equivalence:** Modalities of set $M$ are equivalent for reaching $s'$ from $s$ if it is necessary and sufficient to use any one of the modalities. $M$ is assumed to contain at least two modalities. More formally:

$$
\text{Equivalence} \ (s, M, s') \equiv (\text{Card}(M) > 1) \land (\forall m \in M \ \text{Reach} \ (s, m, s'))
$$

Figure 60: Equivalence formula

Equivalence expresses the availability of choice between multiple modalities but does not impose any form of temporal constraint on them.

![Diagram showing relationships between Complementarity, Assignment, Redundancy, Equivalence, Permanent, Transient, Total, Partial, Device, and Language]

Figure 61: A framework to characterise multi-feature user interfaces with relations between interaction languages, physical devices, and tasks

**10.6.2. ICARE: What is it?**

ICARE is a design environment for specifying and designing multimodal user interfaces based on the CARE properties (Complementarity, Assignment, Redundancy, and Equivalence).
SIMILAR Dreams

Systems support multiple interaction techniques such as the synergistic use of several modalities (speech, gesture, gaze, localisation of a user, etc.). The flexibility that they offer results in increased complexity that current software development tools do not address appropriately. ICARE (Interaction CARE - Complementarity Assignment, Redundancy and Equivalence-) is a component-based approach that allows the easy and rapid development of multimodal interfaces. ICARE is concentrated on input (i.e., from the user to the system), although our model holds for output as well. Nevertheless, we have not tested our approach for output so far.

The ICARE conceptual model includes two kinds of ICARE component:

1. **Elementary components:** Such components are building blocks useful for defining a modality. Two types of ICARE elementary components are defined: Device components and Interaction Language components. We re-use the definition of a modality as the coupling of a physical device d with an interaction language L: \(<d, L>\). A **physical device** is an artefact of the system that acquires (input device) information. Examples of devices include the mouse, microphone, GPS, and magnetometer. An **interaction language** defines a set of well-formed expressions (i.e., a conventional assembly of symbols) that convey meaning. The generation of a symbol, or a set of symbols, results from actions on physical devices. Examples of interaction languages include pseudo-natural language, direct manipulation, and localisation. An **interaction modality** such as speech input is then described as the couple \(<\text{microphone}, \text{pseudo natural language}\, NL>\), where NL is defined by a specific grammar. Similarly graphic input is described in terms of \(<\text{mouse}, \text{direct manipulation}>\).

2. **Composition components:** Such components describe combined usages of modalities and therefore enable us to define new composed modalities. The ICARE composition components are defined on the basis of the four CARE properties - Complementarity, Assignment, Redundancy, and Equivalence -
that may occur between the modalities available in a multimodal
user interface. Composition components can connect from two to
n components and are not dependent on a particular modality.

10.6.3. Elementary Components

Elementary components are dedicated to interaction modalities.
Based on the definition of an interaction modality, we identify two
types of elementary ICARE components, namely Device and
Interaction Language components.

An ICARE Device component represents a supplemental layer
of the physical device driver. For example, the mouse Device
component abstracts the data provided by the mouse driver such
as button pressed/released and movement. Likewise a
microphone Device component abstracts the captured signal into
a recognised utterance while another microphone Device
component abstracts the captured signal into a level of noise. All
ICARE Device components also enrich the raw data from the
device driver by adding information that includes the device’s
operating status, time-stamp, confidence factor of the produced
data, and a description of the device in terms of human
manipulation (passive/active modalities, human actions involved,
and physical location of these actions). An ICARE Device
component is then linked to a listener component, an ICARE
Interaction Language component, in order to form an interaction
modality.

An ICARE Interaction Language component corresponds to the
logical level of an interaction modality. For example, an
Interaction Language component abstracts the data from a
mouse Device component into commands such as the selection
of a menu option. Similarly, another Interaction Language
component (NL component) abstracts a set of characters from a
microphone Device component (recognised utterance) or from a
keyboard Device component into a command. A third example,
shown in Figure 62, corresponds to a passive modality: the 3D
Location component that abstracts data from a localisation
sensor (e.g., GPS) Device component into a user’s location

194
SIMILAR Dreams

expressed in a given coordinate system. These three examples of Interaction Language components underline the fact that such components may need to rely on an external description of well-formed expressions to be obtained. Indeed, to abstract data from the mouse into commands, a description of the graphical interface is required. Likewise, NL recognition requires a description of the pseudo Natural Language to be recognised (NL grammar). Finally, the 3D Location component may require a description of the environment of the user in order to produce an event such as <the user is entering a particular room>. Just as Device components are dependent on the underlying physical devices, Interaction Language components are dependent on a class of Device components that can produce the required inputs. For example, an NL component requires a set of characters as inputs that can for example be produced by a microphone Device component or a keyboard Device component. Finally, like ICARE Device components, ICARE Interaction Language components also enrich the data by adding generic information that include the time-stamp as well as a confidence factor of the produced data.

Device and Interaction Language components are the building blocks for defining modalities. The designer can then combine these components in order to specify a new composed modality, in other words, a combined usage of several modalities.

10.6.4. Composition Components

The CARE properties characterise the different usages of multiple modalities. Based on the CARE properties, we define three composition components: t: Complementarity component, Redundancy component, Equivalence component, and Redundancy/Equivalence component. Assignment and Equivalence are not modelled as components in our ICARE model. Indeed, as shown in Figure 62, an assignment is represented by a single link between two components. An ICARE component "Magnetometer" linked to a single component "3d orientation" implies that "Magnetometer" is
SIMILAR Dreams

assigned to "3d orientation". As for Assignment, Equivalence is not modelled as a component. As shown in Figure 63, when several components (2 to n components) are linked to the same component, they are equivalent. In our previous multimodal systems developed using ICARE components, we explicitly used an Equivalence component that had no functional role (no treatment except defining a new time-stamp for the data) but constituted an aid while manually assembling components. Using our platform under construction that will allow the user to graphically assemble ICARE components, such Equivalence component has no more utility.

In Figure 62 we present an example of ICARE specifications that includes Complementarity components. Let us consider the Complementarity-1 component. In order to compute the location and orientation of the user that is required by the application, two passive modalities are used in a complementary way. The Complementarity-1 component of figure 62 defines a customisable temporal window for merging data received by the two Interaction Language components (respectively orientation in radians and location as latitude/longitude in WGS84 normalization and altitude in meters).

![Figure 1. Example of ICARE specification (the MEMO interaction)](image)

Figure 62. Example of ICARE specification

As ICARE elementary components, ICARE composition components enrich the data by adding generic information that
SIMILAR Dreams

includes the time-stamp and a confidence factor of the produced combined data. In addition composition components include parameters that the designer can fix for customizing the composition mechanism.

10.6.5. ICARE Platform

The ICARE platform enables the designer to manipulate and assemble ICARE software components graphically in order to specify the multimodal interaction dedicated to a given task of the interactive system under development. From this specification, the code is generated automatically. To understand the scope of our ICARE platform fully we show in Figure 63 where the automatically generated code is located within the complete code of the interactive system structured along the ARCH software architectural model.

![Diagram of ARCH software architectural model and ICARE components](image)

Figure 63

The originality of the ICARE platform lies in the fact that it is for designers, not developers. Indeed the user of the ICARE platform selects the modalities and specifies the combination of modalities in terms of the CARE ergonomic properties, all by assembling software components graphically without knowing the details of the components' code. From this high level
SIMILAR Dreams specification, the code of the input multimodal UI is then generated.

Figure 64 presents a sketch of the ICARE platform’s user interface. It contains a palette of components, an editing zone for assembling the selected components, and a customisation panel for setting the parameters of the components. Although the complete ICARE platform is not yet available, we have already designed and developed several components, including modality components, as well as combination components in order to validate our approach.

By assembling these components manually, we have developed several multimodal systems. The following section describes these ICARE components that will in the near future be manipulated graphically in the ICARE platform.

Figure 64. Sketch of the graphical ICARE platform.
10.7. Cameleon

The CAMELEON (Context Aware Modelling for Enabling and Leveraging Effective interactiON) Reference Framework is intended to support the development of context-sensitive user interfaces in a model-based approach. It describes models at four abstraction levels (Figure 65) from task specification to the running interface:

- **The Tasks and Concepts** level describes the interactive systems' specifications in terms of the user tasks to be carried out and the domain objects manipulated by these tasks.
SIMILAR Dreams

- The **Abstract User Interface** (AUI) is an expression of the UI in terms of presentation units, independently of which interactors are available. A presentation unit is a presentation environment (e.g., a window or panel) that supports the execution of a set of logically connected tasks.

- The **Concrete User Interface** (CUI) is an expression of the UI in terms of abstract interactors and their position. The concrete UI is still only a mock-up in the development environment. It can be modified by the designer.

- The **Final User Interface** (FUI) is generated from a concrete UI expressed in the source code of any programming language or mark-up language (e.g. Java or HTML). It can then be interpreted or compiled.

The main contribution of the reference framework is its capability to be instantiated in many ways. This clearly characterises the functional coverage of current tools and makes it possible to make comparisons and identify requirements for future tools. Examples of tools developed in the CAMELEON reference framework are TERESA and Vaquita/ReversiXML. TERESA generates UIs for multiple devices from a single task model. The model is filtered to produce a platform specific task model. This task model is further transformed into an Abstract User Interface (AUI). Vaquita/ReversiXML extracts an abstract description from a UI's code and generates the code of new UIs adapted to the target platform(s).

Drawing upon the reference framework, CAMELEON-RT (Run Time) is the software infrastructure for accommodating the dynamic adaptation of context-sensitive user interfaces in heterogenous information spaces.
SIMILAR Dreams

The architectural model is organised in three levels of abstraction:

The **Interactive Systems layer** consists of the interactive applications that users currently run in the information space.

The **DMP-middleware layer** provides services for modelling the physical space (context infrastructure), supporting the creation of dynamic heterogeneous clusters of resources, and adapting the UI when distribution and migration occur.

The **platform layer** groups the hardware and legacy operating system(s) of an interactive space. The hardware includes surfaces and instruments, computing and communication facilities, and sensors and actuators.

Figure 66 shows CamNote, which is a slide viewer implementing the CAMELEON-RT architecture and running on a dynamic heterogeneous platform. This platform may range from a single PC to a cluster composed of a PC and a PDA. The viewer switches between the platform configurations automatically according to the available devices. The user interfaces are also reshuffled automatically.

![Figure 66](image)

Figure 66. (a) The user interface of CamNote when distributed on a PC and a PocketPC screens; (b) the control panel when displayed on the PC screen.
SIMILAR Dreams

10.8. Modality Theory

To achieve at least part of the understanding needed, it appears that the following objectives defining the research agenda of Modality Theory (Bernsen 1993) should be pursued:

(1) To establish an exhaustive taxonomy and systematic analysis of the unimodal modalities that go into the creation of multimodal output representations of information.

(2) To establish an exhaustive taxonomy and systematic analysis of the unimodal modalities that go into the creation of multimodal input representations of information. Together with Step (1) above, this will provide sound foundations for describing and analysing any particular system for interactive representation and exchange of information.

(3) To establish principles for how to legitimately combine different unimodal output modalities, input modalities, and input/output modalities for usable representation and exchange of information.

(4) To develop a methodology for applying the results of Steps (1) – (3) above to the early design analysis of how to map from the requirements specification of some application to a usable selection of input/output modalities.

(5) To use results in building (possibly automated) practical interaction design support tools.

The research agenda of Modality Theory thus addresses the following general problem: Given any particular set of information that needs to be exchanged between user and system during task performance in context, identify the input/output modalities that constitute an optimal solution to the representation and exchange of that information. This is a hard problem for two reasons. Firstly, already at the level of theory there are a considerable number of unimodal modalities to consider whose combinatorics, therefore, is quite staggering.
SIMILAR Dreams

Secondly, when it comes to applying the theory in development practice, the context of use of a particular application must be taken thoroughly into account in terms of task, intended user group(s), work environment, relevant performance and learning parameters, human cognitive properties, etc.

A particular modality is not simply good or bad at representing a certain type of information – its aptness for a particular application very much depends on the context. This adds to the combinatorics generated by the theory to form an open-ended space of possibilities for consideration by the developer, a space which, furthermore, remains poorly mastered despite decades of HCI (Human to Computer Interaction)/HHSI (Human-Human-System Interaction) research, primarily because such is the nature of engineering as opposed to abstract theory.

The space of unimodal output representations can be carved up at different levels of abstraction. We have seen that already above, in fact, because the three media of graphics, acoustics, and haptics may be viewed as a very general way of structuring the space of unimodal output representations. What will be proposed in the following is a downwards extensible, hierarchical generative taxonomy of unimodal output modalities that at present has four levels: a super level, a generic level, an atomic level, and a sub-atomic level. In terms of the generative steps to be made, the generic level comes first. Thus, the taxonomy is based on a limited set of well-understood generic unimodal modalities. The generic modalities are generated in turn from sets of basic properties.

We generate the generic-level unimodal output modalities from a small set of basic properties that serve as a robust way to distinguish modalities from one another within the taxonomy. The properties are: linguistic/non-linguistic, analogue/non-analogue, arbitrary/non-arbitrary, and static–dynamic. In addition, a distinction is made between the physical media of expression of graphics, acoustics, and haptics, each of which is characterised by a very different set of perceptual qualities (visual, auditory and
SIMILAR Dreams
tactile, respectively). These media determine the scope of the
taxonomy.

Linguistic representations are based on existing syntactic-
semantic-pragmatic systems of meaning. Linguistic
representations, such as speech and text, can, somehow,
represent anything and one might therefore wonder why we
need any other kind of modality for representing information in
HHSI. The basic reason appears to be that linguistic
representations lack the specificity that characterises analogue
representations (Stenning and Oberlander 1991, Bernsen 1995).
Instead, linguistic representations are abstract and focused:
they focus, at some level of abstraction, on the subject matter to
be communicated without providing its specifics.

Analogue representations, such as images and diagrams,
represent aspects of similarity between the representation and
what it represents. These aspects can be many, as in holograms,
or few, as in a standard data graphics pie graph (or pie chart).
Note that the sense of "analogue" in Modality Theory is only
remotely related to that of "analogue (vs. digital)". Being
complementary to linguistic modalities, analogue representations
(sometimes called "iconic" or "isomorphic" representations) have
the virtue of specificity but lack abstract focus, whether they be
static or dynamic, graphic, acoustic or haptic.

The distinction between non-arbitrary and arbitrary
representations marks the difference between representations
that, in order to perform their representational function, rely on
an already existing system of meaning and representations that
do not.

Static representations and dynamic representations are mutually
exclusive. However, the notion of static representation used in
Modality Theory is not a purely physical one (what does not
change or move relative to some frame of reference) nor is it a
purely perceptual one (what does not appear to humans to
change or move). Rather, static representations are
representations that offer the user freedom of perceptual
SIMILAR Dreams

inspection. This means that static representations may be decoded by users in any order desired and as long as desired. Dynamic representations are transient and do not afford freedom of perceptual inspection.

Modality Theory in general and the taxonomy of unimodal output modalities in particular serve the clear and efficient presentation and exchange of information.

The hypothesis that has been confirmed up to this point in the development of Modality Theory and which is inherent to the atomic level of the taxonomy of unimodal output modalities, is a rather strong one. It is that the atomic level fulfils the requirements of completeness, uniqueness, relevance, and intuitiveness stated above.

Any multimodal output representation can be characterised exhaustively as consisting of a combination of atomic-level modalities.

Beyond Literal Meaning. Metaphor and Metonymy

Sometimes it may be preferable to use non-literal meaning instead, or in addition, i.e., to use modalities intending them to be understood in a way which is different from their literal meaning. Metaphoric use of modalities is probably the best known kind of non-literal use in interaction design so far, such as in the static graphic desktop metaphor. In metonymy, a complex subject matter is referred to through some simple part of it.

In general, Modality Theory views non-literal meaning as being derived from literal meaning through subtraction of a smaller or greater amount of the literal connotations of an expression of information in some modality. Modalities can be organised into modality structures, such as lists and tables, and modalities can assume modality roles, such as when being used as icons. We have seen that modalities can have non-literal uses in addition to,
SIMILAR Dreams

or in replacement of, their literal uses. These phenomena are not exclusive. It is perfectly possible, for instance, to make a table of metaphorical icons.

Modality documents define, explain, analyse, and illustrate unimodal modalities from the point of view of interaction design support. The shared document structure includes the following entries:

(1) Modality profile
(2) Inherited declarative and functional properties
(3) Specific declarative and functional properties
(4) Combinatorial analysis
(5) Relevant operations
(6) Identified types -of
(7) Illustrations

Following the research agenda of Modality Theory, we should at this point address the issue of how to combine unimodal output modalities, unimodal input modalities, and unimodal input/output modalities into usable multimodal representations. However, as the taxonomy of unimodal input modalities is not quite ready yet, this issue will be postponed to the final section of the present chapter.

10.8.1. MULTIMODALITY

Getting a theoretical handle on multimodality would constitute a major result of Modality Theory. As this is work in progress, we are not yet able to present any well tested approach that could be claimed to be superior to, or a valuable complement to, the best current approach.

Modality Theory–based Approaches

How might Modality Theory do better than the best current approach? The theory is superior to the empirical approach in that Modality Theory allows complete generation of all possible input, output, and input/output modality combinations at any level,
SIMILAR Dreams

such as the atomic level, and cross-level as well. However, whilst complete generation is possible in a way that is sufficient for all practical interaction design purposes, the combinatorial explosion involved makes it practically impossible to investigate all the generated modality combinations systematically. For instance, if we wanted to investigate all possible n-modal atomic input/output modality combinations where n=10, the number of combinations to be investigated would run into millions. Still, there do seem to be interesting opportunities for exploring this generative/analytic approach by carving out combinatorial segments from the taxonomy for systematic analysis, such as a speech-cum-other-modalities segment, or an input-manipulation-cum-other-modalities segment. These exercises could be further facilitated by tentatively clustering families of similar modalities and treating these as a single modality whose interrelations with other modalities are being investigated. An example could be to treat all analogue static graphics modalities as a single modality, given the fact that these modalities have a series of important properties in common. It is perfectly legitimate to ask questions, such as “How does this particular family of tri-modal combinations combine with other modalities?”

An alternative to the generative approach just described could be to scale up the SMALTO tool to address all possible modality combinations. The problem, however, is that this would be likely to produce lists of hundreds of relevant modality properties, creating a space of information too complex for practical use. Part of the usefulness of SMALTO lies in the fact that SMALTO operates with such a small number of modality properties that it is humanly possible to achieve a certain familiarity with all of them, including the broad implications for interaction design of each them, rather quickly. It might be preferable, therefore, to use the SMALTO approach in a slightly different way, i.e. by producing modality properties for limited segments of combinatorial input/output modality space according to current needs, just like SMALTO itself does.
SIMILAR Dreams

We are currently working on a third approach, which is to turn Modality Theory as a whole into a hypertext/hypermedia tool using a common format for modality representation similar to the format described in Section 3 but with an added entry for modality properties. By definition, the tool would include all identified modality properties. The challenge is to make the tool useful for interaction designers who are not, and do not want to become, experts in the theory, for instance by including a comprehensive examples database. In itself, this tool would not constitute a full scientific handle on multimodality in the sense of a systematic approach to multimodal combinations. However, building the Modality Theory tool does seem to be a necessary next step and one that would also facilitate achieving the ultimate goal of mastering the huge space of multimodal combinations.

Finally, a fourth approach is to analyse the “good compounds” (Section 6.1) in terms of modality properties in order to explore whether any interesting scientific generalisations might appear.

To complete the research agenda of Modality Theory, we need a well-tested taxonomy of unimodal input modalities, a Modality Theory hypertext/hypermedia tool, and exploration of additional ways in which the theory can be of help in achieving a systematic, creative, and predictive understanding of input/output modality combinations, including those that have not yet been widely used, if at all. These themes are topics of the current research work.

10.9. UsiXML

USer Interface eXtensible Markup Language (UsiXML) is a User Interface Description Language issued from the Cameleon project to specify context-aware, multi-platform, multi-modal user interfaces.

Currently, developing the User Interface (UI) of interactive applications is very difficult because of the complexity and the diversity of existing development environments and the high
SIMILAR Dreams

amount of programming skills required by the developer to reach a usable UI, i.e., markup languages (e.g., HTML), programming languages (e.g., C++ or Java), development skills for communication, and skills for usability engineering.

These difficulties are exacerbated when the same UI should be developed for multiple contexts of use such as multiple categories of users (e.g., having different preferences, speaking different native languages, potentially suffering from disabilities), different computing platforms (e.g., a mobile phone, a Pocket PC, an interactive kiosk, a laptop, a wall screen), and various working environments (e.g., stationary, mobile).

Although designers and programmers are involved in these types of project, the available tools mainly target the developer. Therefore, it is rather difficult for a designer to design a UI for multiple contexts of use while avoiding reproducing multiple UIs for multiple contexts of use. This work proposes a way to separate responsibilities in these types of projects.

UsiXML (which stands for USer Interface eXtensible Markup Language) is an XML-compliant markup language that describes the UI for multiple contexts of use such as Character User Interfaces (CUIs), Graphical User Interfaces (GUIs), Auditory User Interfaces, and Multimodal User Interfaces.

In other words, interactive applications with different types of interaction techniques, modalities of use, and computing platforms can be described in a way that preserves the design independently from peculiar characteristics of physical computing platforms.

UsiXML is intended for non-developers, such as analysts, specifiers, designers, human factors experts, project leaders, novice programmers, and so on.

Of course, UsiXML can also be used by experienced developers. Thanks to UsiXML, non-developers can shape the UI of any new interactive application by specifying or describing it in UsiXML, without requiring the programming skills usually found in markup
SIMILAR Dreams

languages (e.g., HTML) and programming languages (e.g., Java or C++).

Figure 67: UsiXML Model collection

UsiXML consists of a User Interface Description Language (UIDL), that is to say a declarative language capturing the essence of what a UI is or should be independently of physical characteristics.

UsiXML describes at a high level of abstraction the elements constituting the UI of an application: widgets, controls, containers, modalities, interaction techniques, etc..

UsiXML allows cross-toolkit development of interactive application.

A UI of any UsiXML-compliant application runs in all toolkits that implement it: compilers and interpreters.

UsiXML supports device independence: A UI can be described in a way that remains autonomous with respect to the devices used in the interactions such as mouse, screen, keyboard, voice recognition system, etc.

If needed, a reference to a particular device can be incorporated.

UsiXML supports platform independence: A UI can be described in a way that remains autonomous with respect to the various computing platforms, such as a mobile phone, Pocket PC, Tablet PC, laptop, desktop, etc.
SIMILAR Dreams

If needed, a reference to a particular computing platform can be incorporated.

UsiXML supports modality independence: a UI can be described in a way that remains independent of any interaction modality such as graphical interaction, vocal interaction, 3D interaction, or haptics.

If needed, a reference to a particular modality can be incorporated.

UsiXML allows re-use of elements previously described in anterior UIs to compose a UI in new applications.

10.10. AMODEUS model for dialogue controller

PAC-Amodeus is a conceptual model useful for devising architectures driven by user-centred properties, including multithreading and multimodality. PAC-Amodeus blends the principles of both Arch and PAC. Arch and its companion, the "slinky" metamodel, provide the appropriate hooks for performing engineering trade-offs such as identifying the appropriate level of abstraction for portability, making semantic repair, or distributing semantics across the components of the architecture. In particular, the five component structure of Arch includes two adapters, the Interface with the Functional Core and the Presentation Techniques Component, that allow the software designer to insulate the key element of the user interface (i.e., the Dialogue Controller) from the variations of the functional core and of the implementation tools (e.g., the X window environment).

The Arch model, however, does not provide any guidance about the decomposition of the Dialogue Controller, nor does it indicate how salient features in new interaction techniques (such as parallelism, fusion, and fission of information) can be supported within the architecture. PAC, on the other hand, stresses the recursive decomposition of the user interface in terms of agents,
SIMILAR Dreams

but does not pay attention to engineering issues. PAC-Amodeus combines the best of the two worlds.

Succinctly, the five components of the Arch defines the levels of abstraction appropriate for performing engineering trade-offs such as setting the boundaries between the levels of abstraction. We offer the notions of physical device and interaction language as criteria for setting these boundaries. For example, the designer may decide that the Low Level Interaction Component is device dependent. At a higher level of abstraction, the Presentation Techniques Component is device independent but language dependent. At the top of the Arch, the Dialogue Controller is both language and device independent.

PAC-Amodeus refines the Dialogue Controller into a set of co-operative agents that capture parallelism and information processing (e.g., data fusion) at multiple levels of abstraction. In turn, an agent is modelled as a three facet structure:

1. the Presentation facet is in direct contact with the Presentation Techniques Component of the Arch. It can be used to implement extensions on top of the Presentation Techniques Component;

2. the Abstraction facet is in direct contact with the Interface with the Functional Core;

3. the Control facet manages the links and constraints between its two surrounding facets (i.e., the Presentation and the Abstraction facets) as well as its relationships with other agents. As in ALV [8], the Control facet provides the hook for expressing constraints between different perspectives of the same concept.

In combining the Arch principles with PAC, one obtains an "engineerable" model that supports properties inherited from the agent paradigm. Figure 1b illustrates the application of PAC-Amodeus to the software design of MATIS. The Functional Core hosts the database of American cities, airline companies, flight numbers, departure and arrival times, etc. SQL requests are required to access information stored in the database. The
SIMILAR Dreams

Interface with the Functional Core (IFC) operates as a translator between the SQL formalism and the data structures used in the Dialogue Controller. In MATIS, the IFC serves as a communication bridge. It can also be used to restructure conceptual objects in a form suitable for the purpose of the interaction.

The Dialogue Controller (DC) is organised as a two-level hierarchy of agents. This hierarchy has been devised using heuristic rules. For example, because requests can be elaborated in an interleaved way, there is one agent per pending request.

At the other end of the spectrum, the Low Level Interaction Component (LLIC) is instantiated as two components inherited from the underlying platform: (1) The NeXTSTEP event handler and graphics machine, and (2) the Sphinx speech recogniser, which produces character strings for recognised spoken utterances. Mouse-key events, graphics primitives, and Sphinx character strings are the interaction objects exchanged with the Presentation Techniques Component (PTC).

In turn, the Presentation Techniques Component (PTC) is split into two main parts: the graphics objects (used for both input and output) and the NL parser (used for input only). Graphics objects result from the code generation performed by Interface Builder. The Sphinx parser analyses strings received from the LLIC using a grammar that defines the NL interaction language. As discussed above, the PTC is no longer dependent on devices, but processes information using knowledge about interaction languages.

10.11. References

Reliability ordered cascade

SIMILAR Dreams

Multimodal emotion detection (prosody + facial)


Mutual information in multimodal SP


Multimodal biometrics


Multimodal medical image registration
du Bois d’Aische Aloys, De Craene Mathieu, Grégoire Vincent, Macq Benoit, Warfield Simon, 'Efficient Multi-modal dense field non-rigid registration: alignment of histological and section images,' Medical Image Analysis, ITK special issue, 2005

ICARE
Coutaz, J.I., Nigay, L., Salber, D., Blandford, A., May, J., and Young, R.M., Four Easy Pieces for Assessing the Usability
SIMILAR Dreams

of Multimodal Interaction: t: the CARE Properties, in

J. Bouchet, B. Mansouxi and L. Nigay, A Component-based
Approach: ICARE, ACM-CHI’2005 Workshop on The
Future of User, Interface Design Tools, Portland, USA,
April 2005.

J. Bouchet, L. Nigay, ICARE: A: a Component-Based Approach
for the Design and Development of Multimodal Interfaces,
Extended Abstracts of CHI’04, Vienna, Austria, April 2004.
pp. 1325-1328

S. Dupuy-chessa, L. du Bousquet, J. Bouchet, Y. Ledru, Test of
the ICARE platform fusion mecha-nism, 12th International
Workshop on Design, Specification and Verification of

CAMELEON

Website: http://giove.cnuce.cnr.it/cameleon.html

Mori, G., Paternò, F., Santoro, C. Tool Support for Designing
Nomadic Applications Proceedings of IUI 2003 -Miami,

L. Bouillon, J. Vanderdonckt, N. Souchon, Rétro-ingénierie du
modèle de présentation pour les pages Web, Revue

Paternò, F. Santoro, C. A Unified Method for Designing
Interactive Systems Adaptable to Mobile and Stationary
Platforms. Interacting with Computers. Vol. 15, No. 3, pp

G. Mori, F. Paternò, C. Santoro, Design and Development of
Multi-Device User Interfaces through Multiple Logical
Descriptions, IEEE Transactions on Software Engineering,

L. Balme, A. Demeure, N. Barralon, J. Coutaz, G. Calvary,
CAMELEON-RT: a: a Software Architecture Reference
Model for Distributed, Migratable, and Plastic User
Interfaces, EUSAI 2004, LNCS 3295, Markopoulos et al.,
SIMILAR Dreams

Modality theory


UsiXML
Website: http://www.usixml.org


AMODEUS
Website: http://www.mrc-cbu.cam.ac.uk/amodeus/

