

# QUEVICO QA MODEL FOR VIDEO-BASED INTERACTIVE MEDIA

*Hidekatsu Izuno, Yuichi Nakamura, and Yuichi Ohta*

IEMS, University of Tsukuba  
1-1-1 Tennodai, Tsukuba 305-8573, JAPAN  
yuichi@image.esys.tsukuba.ac.jp

## ABSTRACT

This paper introduces the basic idea of QUEVICO, its tagset, answering process, and our prototype system. QUEVICO is a question-based video composition scheme in which video indexing and editing is designed from the viewpoint of “question and answer”, and in which multi-view videos can be effectively used. Based on the tagset in this framework, we can index a video in a suitable form for answering typical questions. The system retrieves a relevant video portion by this framework, even if a complete set of indices are not given. By editing and arranging the retrieved portion, a smart answer will be given to the user.

## 1. INTRODUCTION

If a person asks us to teach how to cook a *sashimi*, we strongly need visual explanation, *e.g.*, a picture of a raw fish, a demonstration for cutting a fish, and so on. For realizing such explanations on interactive media, we often need videos in addition to a text or a speech. The aim of this research is to create video-based interactive media that give comprehensible answers to typical questions on such works.

While many works have been reported on intelligent help systems or question-answering systems that can communicate in natural languages, they are not directly applicable to the video-based media. There are particular problems on handling videos for the purpose of question-answering.

For this purpose, we propose a novel framework *QUEVICO*<sup>1</sup> that is designed for realizing intelligent video-based teaching materials. In the following sections, we will present the basic idea of QUEVICO and interactive video-based media, the composition of data, and the mechanism for answering questions.

## 2. QUEVICO

One important advantage of using videos is the richness of information. Videos can give different kinds of informa-

<sup>1</sup>“QUEstion-based Video COMposition”. In Japanese myths, QUEVICO (or KUEBIKO) is a god of knowledge, whose figure is a scarecrow and who is a guardian of agriculture.

tion simultaneously. For example, “How much should I cut it?” may mean “How long ...?”, “With which kitchen knife ...?”, “When ...?”, and so on. For answering this question in natural language, it is necessary to estimate the category of requested information and to compose sentences given as an answer. This may require precise understanding of the user’s intention, or thorough search, in the stored knowledge, for all possible answers.

On the other hand, a video that captured the cutting action can give all together the information on “how long”, “how much”, “with which tools”, and so on. What we have to do is *to know which portion of a video is the relevant answer, or which portion of a video potentially has the information the user can draw an answer.*

A video, however, does not hold complete information of the scene. A cameraman or a director carefully chooses a camera position, a view field, and carefully edits the obtained video. A director often edits out portions that he/she do not want to show. This process determines what information is kept in the video and what can be easily grasped at a glance. Therefore, when we explain something with a video, we need *to use a video taken with an appropriate setting and camerawork.* In this sense, it is desirable that we have multi-view videos without editing.

With the above conditions, the videos can be good resources that reduce difficult for answering questions.

### 2.1. Answering by Multimedia

Figure 1 shows the rough idea of a typical video-based interactive media. Video data are stored and structured by tagging. Through the interaction between a user and the system, the system estimates which information should be given to the user, and gives an answer with retrieved video fragments.

In constructing such media, we have to deal with the following problems:

**media/modality selection:** A video contains still images, moving images, sound, speech, time, and so on. However, it is not well investigated which media/modality is most appropriate for answering a question.

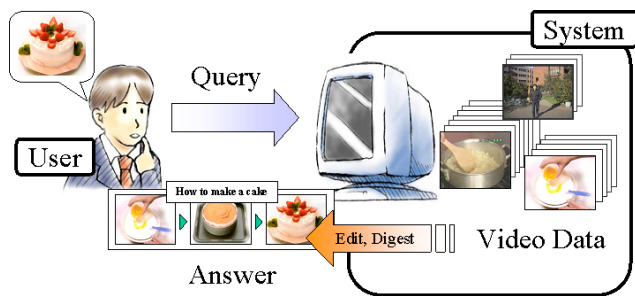


Fig. 1. Video-based interactive media

**handling ambiguous and compound information:** A question is sometimes ambiguous, or requests a compound of information. A simple pinpoint answer is often inappropriate. On the other hand, a video can give a variety kinds of information simultaneously. Such compound of information should be well considered in handling multimedia.

These are intrinsic characteristics of multimedia, and we always have to handle variety of compound information. For this purpose, we designed QUEVICO as a novel framework for dealing with these intrinsic problems of multimedia.

## 2.2. QUEVICO's Features

QUEVICO has the following two important features:

- Video indexing and editing is designed from the viewpoint of “question” and “answer”. A variety of questions were considered, and a XML tagset for marking-up each portion that potentially gives an answer to those questions was determined. The answer is chosen by considering “what information is requested by a question” and “which is the best method to show relevant data for the requested information”. Moreover, we designed a QA model that works with incomplete or poor tagging.
- Multi-view videos without editing are effectively used. When we deal with edited videos such as TV programs, they are insufficient since essential information is often edited out. By dealing with multi-view videos without editing, we simplify the problem of selecting and editing video portions.

Thus in the model shown in Figure 1, video data are stored and marked-up by the tagset of QUEVICO. They are taken by multiple cameras and stored without editing. For the interaction between a user and the system, we are currently using a simple process that matches between an ac-

Tab. 1. Typical questions

How can I make a sashimi?  
 How should I cut it?  
 What kind of food do I need to prepare?  
 Why should I add water?  
 Is there any suggestions?  
 Which kind of fish is suitable for this dish?  
 How much sugar do I need to put?  
 How is the finish form?  
 How would a professional cook do?  
 How long does it take?  
 To which shape do I need to cut?  
 Salt is running out. What should I do?

tual question and “question type” with other required values for answering <sup>2</sup>.

## 2.3. Related Work

Many works have been reported on video indexing and retrieval, *e.g.*, Informedia project[4], and they introduced various methods for analyzing and structurizing videos. One of the most common ways for video retrieval is to search for significant words from transcripts, and another is to find relevant video segments in terms of color features. Such kinds of video retrieval, however, are methods of “retrieving related data portions”, and is not of “answering questions”. In this sense, our approach, that is video management based on question, is unique. Moreover, our framework uses multi-view videos in order to compose comprehensible answers.

As for tagging, although the MPEG-7 standard contains semantic description, we currently use our original tagset in XML, since the de facto standard is not ready. We will move to MPEG-7 after the de facto standard appears, and we expect that migration will be easy because XML is incorporated in the MPEG-7 standard.

In the natural language processing and AI field, many researchers have reported their interactive systems, some of which are used for question-answering systems (for example, [1] gives good pointers). Our research is different on the point of concentrating on video specific problems, such as video tagging, editing, and the selection of multimedia data. Hopefully, useful techniques of natural-language-based interaction schema can be incorporated into our conversational module.

## 3. QA MODEL IN QUEVICO

### 3.1. Question-based Tagset

We intensively gathered broadcasted cooking shows and made a list of possible questions. Table 1 shows a portion of the

<sup>2</sup>We do not focus on the natural language processing, since we want to concentrate on the problem on handing videos.

**Tab. 2.** Typical questions and requested information

Question type	Requested information
Tell me how to (verb)	task, dependency, duration
What should I (verb)?	task, substitution, instrument, patient, dependency
Why do I need to (verb)?	reason, dependency, output
What happens when I (verb) it?	output, method
What should I use?	material, substitution, input/output, reason
How many/much do I need to (verb)?	degree, duration, input-quantity, method, task
Is there anything to pay attention?	note, method, degree, quantity
How will be the result?	input/output, task
Who is (verb)+ing?	agent, location, dependency
What is he/she (verb)+ing?	patient, instrument, state, reason, method
Where is he/she (verb)+ing?	location, task, agent, destination

collection that amounts to around 300, which is almost saturated in our preliminary experiments.

By analyzing them, we found two important features:

- If we prepare around 30 prototypes of question, they cover the majority of possible questions.
- Most of questions concern and requests information of tasks or objects.

Table 2 shows examples of categorized questions and the information that requested by them. The first column shows prototypes, and the second column shows the information that each type of question requests. Those patterns are also common in other areas of instruction or teaching videos that explain “*how or what to do*”.

Based on the above idea, we devised the tagset for marking-up the potential answers to a question. Tags for defining data segments are simple. Physical portions of a video, *e.g.*, areas (regions) in a image, video segments, are marked-up, and they can have attributes for describing them. For more abstract portions of a video, we have tags for a “task”, or for an “object”. Those tags can be directly attached to the video data, or they can be attached to a scenario or meta-data if they exist.

Figure 2 shows a simple example of a tagged description. Here, a tag pair for a task (<task> and </task>) specifies tasks performed in a video. A task is represented by its name and possible attributes as shown in Table 4. A set of tasks is structurally organized based on the orders of the tasks, and we call the structure as “task tree”. In Figure 2, Two objects are denoted by <object>. An object is represented by the tag as shown in Table 5. Video segments are described by <video-segment> whose “stime”

**Tab. 3.** Questions by the subjects

	# of questions	rate (%)
able to categorize	192	92.9
unable to categorize	32	7.1
total	224	100

represents start time of the segment, “etime” represents the end time. Those tags are referred by one another by their “id”s, such as “t1”, “v1”, and so on.

Note that any of the attributes except “id” and “name” can be omitted. If an attribute value corresponding to required information is directly given by a tag, it will be used as an answer. Otherwise, candidates for an answer are searched by using the scheme described in the next section.

For validating the categories of questions and the tagset, we asked 10 people to fill out questionnaires concerning videos. The subjects are shown a text or both a text and a video on cooking shows, and they are asked to write down questions that they had during reading or watching.

Table 3 shows the statistics of the obtained result. We gathered total 224 questions, and 92.9% of the questions can be categorized in our framework and the answers can be marked-up by our tags. The rest 7% of the questions generally require intelligent processes that are not supported in our system. For example, although we can easily categorize a question “How should I cut?”, it is currently difficult to process a question “Should I cut vertically or horizontally?” even if they share the same answer. This problem is left for future works.

### 3.2. Question Answering

Figure 3 shows the outline of our QA model. Here we use  $Q$  for representing a set of question types, and  $DS$  for a set of data types. Based on this model, the system searches for relevant data segment  $ds_m$  as an answer for question  $q_i$ , and presents it after certain editing.

This search is directed by two different methods:

**Direct method:** This method uses the paths through tags shown in the upper portion of Fig. 3. Since we developed the video indexing based on questions and answers, if enough tags are added to the video data, an appropriate data segment is delineated by following the links given by tags.

**Indirect method:** This method uses the path shown in the lower portion of Fig. 3. This path enables answering even when sufficient tags are not given beforehand. By considering intermediate notions, that is, a set of explanation forms  $F$  and a set of data types  $DT$ , the system obtain a data segment that is not far

```

<iimd>
<video-set>
  <video-segment id="v1" src="cakel.mpg" stime="10s" etime="62s" />
  <video-segment id="v2" src="cake2.mpg" stime="67s" etime="90s" />
</video-set>
<speech-set>
  <sentence>Please bake the cake <span id="p1">until it starts to brawn</span>.
  </sentence>
</speech-set>
<object-set>
  <object id="o1" name="cake" />
  <object id="o2" name="fresh cream" />
</object-set>
<task-set>
  <task id="t1" name="cook" output="#o1">
    <task id="t2" name="bake" patient="#o1" method="#v1" degree="#p1" />
    <task id="t3" name="make up" patient="#o1" input="#o2" method="#v2" />
  </task>
</task-set>
</iimd>

```

Fig. 2. Tagging example

Tab. 4. Attributes of the tag for a task

attribute name	description
id	identifier
name	the name of a task
agent	the agent of the action in a task
patient	the objects of the action in a task
input	the input of a task
input-quantity	the quantify of the input
output	the output of a task
output-quantity	the output-quantity of a task
instrument	tools or materials required for performing a task
location	the location where the task is performed
source	the starting point (location) of the action in a task
destination	the end point (location) of the action in a task
time	the time when the task is performed
degree	the degree or the extent which a task is performed
reason	the reason for performing a task is necessary
substitution	alternative tasks that can substitute a task
note	something to pay attention for performing a task
duration	time length necessary for performing a task
dependency	dependence on other tasks

from the correct answer. This scheme is described in Section 3.3.

Tab. 5. Attributes of the tag for an object

attribute name	description
id	the identifier of an object
name	the name of an object
description	the description for an object
state	the current state of an object
color	the color of an object
shape	the shape of an object
quantity	the quantity of an object
smell	the smell of an object
reason	the reason for requiring an object
substitution	the substitution of an object

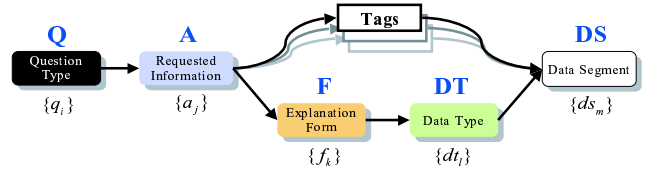


Fig. 3. A multimedia QA model

### 3.3. Indirect Method in QUEVICO

We often need to use videos with incomplete and sparse tags or need to use video data tagged for other purposes, since precise tagging requires much cost.

In dealing with this problem, a video has good characteristics. A video contains rich information, and there is a considerable possibility that the information requested by a question lies in a video, and they can be recognized or drawn by a human.

Suppose a situation where a user asks ‘‘How much do I cut bonito?’’, we can answer by using a still image that shows the slices of bonito after cutting, or by using a video

**Tab. 6.** Example of explanation forms

name	the target’s name that can be person’s name, object name, task name, etc.
appearance	image of an object, image of a person, image for explaining location, etc.
movement	target movement, locus, etc.
adjacent object	an object that is always accompanying the target
input/output	input/output of an operation (task)
composition	part(s) that compose a target

segment that contains cutting motions. However, if no tag which specifies the “degree” of cutting is given, we do not have clear links to delineate which portions of data would be a good answer.

For this purpose, we propose the following QA model shown in Fig. 4. The model has three-stage linking considering the following three types of relations: a relation between each question-type and each requested information type; a relation between each explanation form and each requested information; a relation between each explanation form to each data types. Each element in the model, *e.g.*, *i*-th requested information, has many to many links to other elements. By traversing the relations among these elements, we obtain virtual paths from questions to data segments.

With these notions, we can regard the inner structure of the QA model as follows. Direct product  $Q \otimes A$  represents “which information  $a_i (\in A)$  is requested by each question  $q_j (\in Q)$ ”, which is partially shown in Table 2. We can consider that the value of each matrix element represents the relevance. Similarly, direct product  $A \otimes F$  represents “which explanation form  $f_i (\in F)$  is suitable for giving information  $a_j (\in A)$ ”, and direct product  $F \otimes DT$  holds the relation between an explanation form and a type of data portion. Examples of explanation forms are shown Table 6 and examples of data types are shown Table 7.

By using the above model, we can denote the answering scheme as the following.

$$\text{indirect answering scheme} = Q \otimes A, A \otimes F, F \otimes DT$$

### 3.4. Setting up the Links in QA Model

In the above QA model, we have many parameters such as the elements in  $A \otimes F$  and  $F \otimes DT$ . Although we can roughly estimate those values, it is tough to precisely determine them. For this purpose, we first give rough estimation manually, then adjust it by a neural network training technique. The teaching sample is taken from the precise tagging results that we manually prepared for relatively small data. On the other hand,  $Q \otimes A$  can be easily estimated, we do not use a neural network for this purpose.

First, we give a relevance value to each link, *e.g.*, a path from  $a_i$  to  $f_j$ . The value is between 0 and 1, for which

**Tab. 7.** Example of data types

image region	an image area, <i>e.g.</i> , bounding box, that has the target’s figure.
video segment (scene view)	wide-angled working space view (or a establishing shot)
video segment (agent view)	a person’s view that mainly shows his/her face
video segment (patient view)	close-up view of an object or a patient
video segment (action view)	close shot for capturing a person’s hand movements
audio segment	audio data in a video
word in a speech	a word in a speech, a word in a transcript
task in a scenario	a task description in a tagged form

1 expresses the most tight relationship. Then, the value is scaled to fit a sigmoid function, *e.g.*, between -10 and 10, and given as the initial weight.

Then, samples of input and output of the network are taken from the precise tagging results, and they are used for training network. Suppose that an attribute and its value ( $name_i, value_i$ ) are given in a tag. We regard  $name_i$  as  $q_i$ , and regard the data type of  $value_i$  as  $d_j$ . Training process by backpropagation adjusts the relevance (weight) of each link (connection).

## 4. ANSWERING SCHEME

This system retrieve data segments by the above QA model. The process is composed as follows:

1. The system receives a question form the user. By simple pattern matching, the system determines the type of the question. By using the the words in the question and current status of the system, the system delineates for which task or for which object the user is requesting information.
2. For the strongest requested information which has the largest value, the system searches for the direct answer that is given as the attributes of a tag.
3. If no direct answer is given, potential answers are searched for based on the indirect method. Retrieved data are scored by the relevance of linking. If an element is given scores through two or more different paths, the summation of the scores is considered as the element’s score. Eventually, the data type with the highest score is chosen for the answer.
4. Actual data segment (duration) is chosen by the compatibility between the set of requested information  $\{a_i\}$  requested by the question and the tagged data segment  $\{ds_j\}$ . The set of weights for  $A \otimes F$  links are used for calculating this compatibility.

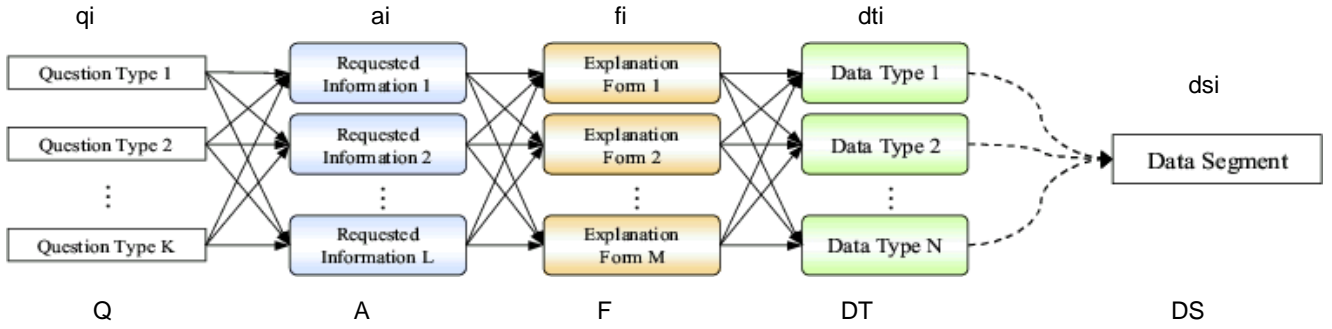


Fig. 4. A network of the QA model



Fig. 5. Multi-view videos (how to cook lightly roasted bonito)

This process effectively uses the rich information of videos. Even if enough tags are not added or an exact answer is not contained in the video data, we can obtain an answer not far from the correct answer. Suppose that a user asks the question about an object, *e.g.*, object’s color or shape. Although one of the best answers is the textual description such as “blue” or “square”, a video clip that captured the object with close-up view can also be a good answer. In this case, we only need to know which view is the object’s close-up. In another example, if a user asks “How long do I need to bake ... ?”, a video fragment implicitly gives an answer by its length, even if no exact answer is given in the video.

Thus, our scheme improves the effectiveness of question-answering mechanism, since we cannot usually add a tag to every detail of video data.

## 5. EXPERIMENTS

Here we shows some examples obtained by our prototype system. The video contents are about cooking, one of which

Tab. 8. Criterion for subjective evaluation

score	explanation
5	An appropriate and relevant answer.
4	Contains a relevant answer.
3	A correct answer can be derived form the data, though it is not an exact answer.
2	A wrong answer.
1	A wrong answer, and it even misleads to a wrong idea.

is “How to cook lightly roasted bonito”. The videos are taken in terms of four views as shown in Fig. 5: scene view (wide-angled establishing shot), agent view (middle shot of a speaker), patient view (close-up shot at objects), and action view (close-up shot of the hands and manipulation). The speech text, that is a transcript, and the scenario along which the video is taken are attached to the video data. Tags are manually added to this combination of data.

An example of questions and the system outputs are presented in Fig. 6. As we can see here, the answers by the system are satisfactory for a simple question. The system is still under development, and more intelligent functions will be added in the near future.

We had preliminary subjective evaluation for verifying the QA model. Answers obtained by using the direct and the indirect method were presented to 10 people. Two examples of a question and an answer are shown in Figure 7. The subjects were asked to evaluate the relevance of an answer, and they were asked to rate it according to the Table 8.

The preliminary result shown in Table 9 is satisfactory. The average score are 3.8 and 3.6 for in direct method and in indirect method, respectively. This means the retrieved and presented data segments are almost the correct answers or the data from which we can draw the correct information.





Fig. 6. Output of our prototype system

Tab. 9. Subjective evaluation

	average	min	max
direct method	3.8	3.2	4.2
indirect method	3.6	2.9	4.0

## 6. DISCUSSION

This research introduced a novel QA model for multimedia question answering. As shown in the above experimental results, we verified good potential of our QA model. However, our model is still incomplete in the following points:

One is automatic indexing. Although we can consider

several works for indexing cooking videos or other types of videos, we need actual experiments for integrating with such works. Another approach is constructing automated video production system[7],[10] that acquires important information through video recording simultaneously.

Another point is question analysis. Our system recognizes question types by simple pattern matching, and it is obvious that we need more sophisticated natural language processing. In the above model, we assumed that the system can easily delineate for which task or object the user is asking a question. This is not true in many cases. We need further investigation for interaction in natural language to identify a topic.

Through both approaches, we will be able to effectively

Q2. How is the bonito prepared?

Q2.かつおはどのような状態なのか教えてください

A.

「事前に焼き付けて氷で冷やしておいたかつお」

Bonito roasted and chilled by ice beforehand.

Q3. What kind of tool do I need to cut bonito?

Q3.かつおを切るにはどんな道具が必要ですか

A.




Fig. 7. Example of data presented in subjective evaluation

handle rich multimedia information for the purpose of question answering.

## 7. CONCLUSION

In this paper, we proposed a novel framework QUEVICO for video-based interactive image media that realizes question-answering as a teacher does. We are currently developing a prototype system based on QUEVICO. Although the implemented functions on this system are still simple, the system showed good potential for answering relatively simple questions.

For future works, we still need intensive work to develop the prototype system, and we will need systematic evaluation in order to prove the effectiveness. We also need to add some important mechanism, for example, a function to recognize the user's status or situation.

## 8. REFERENCES

- [1] The Text Retrieval Conference (TREC, <http://trec.nist.gov/>), Video Retrieval Evaluation Track (<http://www-nlpir.nist.gov/projects/t01v/>)
- [2] J. Marti'nez, "Overview of the MPEG-7 Standard" ISO/IEC JTC1/SC29/WG11 N4509 Pattaya, 2001
- [3] M. Murata, M. Utiyama, and H. Ishihara, "Question Answering System Using Similarity-Guided Reasoning" (in Japanese), Natural Language Processing, pp.135-24, 2000
- [4] H. Wactlar, T. Kanade, M. Smith, and S. Stevens, "Intelligent Access to Digital Video: The Informedia Project", IEEE Computer, Vol.29, No.5, 1996
- [5] M. Smith and T. Kanade., "Video Skimming and Characterization through the Combination of Image and Language Understanding Techniques" Proc. IEEE CVPR, 1997
- [6] H. Jiang and A. Elmagarmid, "WVTDB - A Semantic Content-Based Video Database System on the World Wide Web", IEEE Trans. on KDE, vol.10, NO.6, 1998
- [7] M. Ozeki, Y. Nakamura, and Y. Ohta, "Camerawork for Intelligent Video Production —Capturing Desktop Manipulations", Proc. Int'l Conf. on Multimedia and Expo, 2001
- [8] M. Ozeki, M. Itoh, Y. Nakamura, and Y. Ohta, "Tracking Hands and Objects for an Intelligent Video Production System", Proc. Int'l Conf. on Pattern Recognition, Vol.III, 2002
- [9] M. Murayama, H. Izuno, Y. Nakamura, and Y. Ohta, "Video Icon Diagram: Representation of Video Contents Structure"(in Japanese), IEICE, SIG-PRMU-2001-45, 2001
- [10] M. Ozeki, Y. Nakamura, and Y. Ohta, "Human behavior recognition for an intelligent video production system," IEEE Proc. Pacific-Rim Conference on Multimedia, pp.1153-1160, 2002.