Tile-based Panoramic Live Video Streaming on ICN

Atsushi Tagami*, Kazuaki Ueda*, Rikisenia Lukita†, Jacopo De Benedetto‡, Mayutan Arumaithurai‡,
Giulio Rossi§, Andrea Detti§ and Toru Hasegawa¶
* KDDI Research Inc., Saitama, Japan
† KOZO KEIKAKU ENGINEERING Inc., Tokyo, Japan
‡ University of Göttingen, Göttingen, Germany
§ University of Rome "Tor Vergata", Rome, Italy
¶ Osaka University, Osaka, Japan

***Abstract*—Information-centric networking (ICN) is a future Internet architecture that makes it possible to effectively use various in-network functions, such as cache storage, computing resources and multi-path. However, applications need to be carefully designed in order to fully gain the benefits of these functions. This paper presents a 360-degree panoramic live video streaming application as an example of an application design suitable for in-network functions, especially cache storage. The key ideas are content sharing by video frame tiling and load balancing by transcoding. The implementation based on the application design shows the benefits and the problems of these features. Additionally, this paper discusses problems that remain to be resolved.**

***Index Terms*—Information Centric Networking, Panoramic Video**

I. Introduction

360-degree panoramic video is becoming popular due to its entertainment value, future prospects and adaptive flexibility. Its key feature is that users can navigate the video to areas in which they are interested. This interactivity expands the domain of its applicability, such as camera surveillance, equipment inspection, or live streaming for sports. These features are achieved by the high-spatial resolution, but video streaming requires a high bit ratio. Cisco [1] forecasts that the traffic generated by virtual reality (VR) is expected to grow 11-fold from 2016 to 2021. Thus, the efficient delivery of panoramic video is an important issue for the further improvement of its applicability.

The information-centric networking (ICN) concept is a significant common approach of several future Internet research activities [2]. The approach leverages in-network caching, multi-party communication through replication, and interaction models decoupling senders and receivers. This has the potential to solve the problem of large content distribution. Additionally, edge computing, including fog and mobile edge computing, is becoming an important network component due to its outstanding performance that includes low-latency, reliability, and efficiency. It introduces an intermediary component with processing and storage resources into the network path. An edge node requires high affinity ICN features, such as the location-independence of content and name-based flexible routing.

The work for this paper was performed in the context of the Horizon2020/NICT EU-JAPAN ICN2020 project Grant Agreement No. 723014 and Contract No. 184.

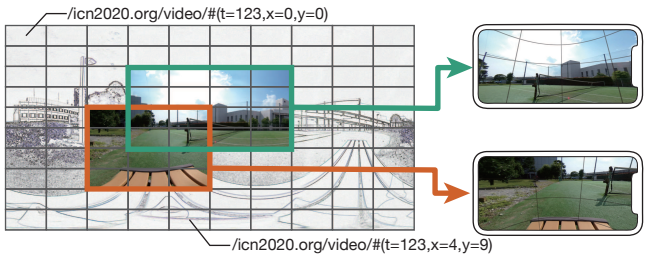

Fig. 1. Tile-based panoramic streaming: Each tile has a unique name and consumers get tiles in their field of views.

However, applications need to be designed while taking the features of the future Internet architecture into consideration. We have studied and demonstrated efficient panoramic live video streaming on ICN [3], [4]. Its design focuses on the effective leveraging of in-network resources, such as storage and computing. In particular, it divides a video frame into several tiles so that more consumers can share them, and it puts the transcoder near consumers to reduce the traffic load on both of the access and core networks. This paper describes the application design in detail and evaluates its benefits and problems.

The rest of this paper is organized as follows. Section II explains the design of our panoramic live video streaming application. Section III describes the implementation and the protocol between the producer and the consumer. Section IV evaluates the benefits of the future Internet features and Section V discusses remaining problems. Section VI covers related work and Section VII concludes this paper.

II. Application Design

A. Tile-based Panoramic Streaming

Panoramic video requires high-spatial resolution, but a user views only a part of the whole video frame. To reduce bandwidth usage, the server can encode only the requested area for each user. However, preparing an encoder for each user requires a massive amount of comupter processing. It is well known that a tile-based streaming technique can avoid

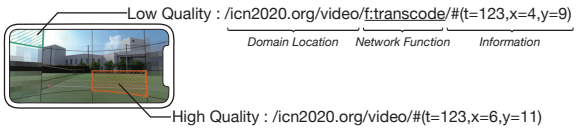

Fig. 2. Multi-level quality tiles and naming scheme: A consumer requires the low-quality tiles where a user does not focus attention.

this scalability problem [5], [6]. The captured video is subdivided into non-overlapping tiles and each tile is independently compressed using a video encoder, e.g., MPEG4, H.264/AVC, or Motion JPEG. A client requests only the necessary tiles according to its desired field-of-view. This technique can lower the complexity of video encoding and reduce the processing delay.

Tile-based streaming, which is based on the retrieval of multiple content objects, is a technique that can be efficiently combined with ICN [3]. Figure 1 shows an example of *named tile-based panoramic streaming*. Each tile is a short video data identified by a unique name which consists of a time sequence number and a coordinate, e.g. /icn2020/livevideo/tokyo/#(t=123,x=2,y=4). A consumer plays the video stream by acquiring tiles having consecutive time sequence numbers, the same method that employed in MPEG-DASH. Consumers issue their interest for the minimum number of necessary tiles to cover their field-of-view. Since each tile is a content object of ICN, it can be cached at intermediate routers and natively delivered in a multicast manner. Thus, even if there are huge number of consumers, the producer only have to transfer all the tiles at once. Named tile-based panoramic streaming can reduce redundancy in both the process and forwarding costs.

B. Multi-level Quality Tiles and Edge Transcoding

Selecting different quality tiles to build a viewport frame is proposed as a way to reduce bandwidth usage without degrading QoE [7]. Figure 2 shows an example of multi-level quality tiles. The user is not cocerned about the quality of the tiles for areas where he does not focus his attention, e.g., outside or at the edge of sight. Thus, the producer provides some quality levels (e.g., high and low), and consumers require tiles of appropriate quality according to the user's field-of-view. However, the multi-level quality technique increases the number of available ICN content objects for the same tile, since a content object is generated for each quality level. Therefore, when different users request the same tile but with a different quality, each request is processed separately. These occurrences cause a reduction in the cache hit ratio, and as a consequence the volume of traffic may increase.

To cope with this problem, we introduce edge transcoding implemented on the in-network computing resource. The key idea is that the producer generates only tiles of the highest quality and either the producer or an intermediate node generates the other lower-quality tiles from those titles. We use a keyword-based naming scheme [8] to develop this idea. As Fig. 2 shows, the name consists of the domain

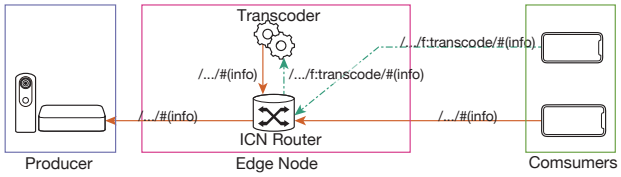

Fig. 3. An example of Interest message flows: The domain location is omitted. "#(info)" means an identifier of a tile, e.g., #(t=123, x=6, y=11).

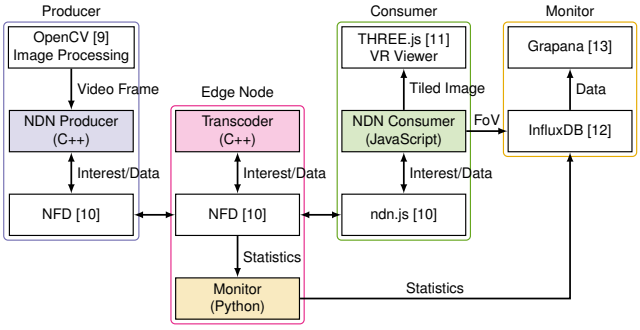

Fig. 4. System overview: The colored boxes mean components which we developed. Others are open source applications and libraries.

location, network function and information. The name function expressed in "f:transcode" requests the tile of the highest quality associated with the information and coverts it into a tile of lower quality. This generated tile is identified by the name including the network function, and also receives the benefits of ICN, i.e., in-network caching and multicast-like forwarding.

Figure 3 shows an example of Interest message flows with edge transcoding. The producer provides tiles having two levels of quality (high and low) identified by the presence or absence of the named function. The transcoder is installed in the intermediate ICN router called an *edge node*. The router forwards Interest message which has a named function to the transcoder. The transcoder requests the high-quality tile and converts it into a low-quality tile. The benefits of this edge transcoding with ICN features are as follows:

Improving cache hit ratio: Since only tiles of highest quality are forwarded between the producer and the edge node, the cache hit ratio is not affected by the increase in the type of tiles.

Request classification: Comparing with proxy-based approaches on the current IP network, ICN-based approaches can retrieve necessary requests for the transcoder without the application-level packet inspection.

Fault tolerance: Even if the transcoder fails, Interest packets are forwarded to the producer and the service is continued.

III. Implementation

A. System Composition

We implement named tile-based live panoramic streaming on NDN. Figure 4 provides an overview of our system in-

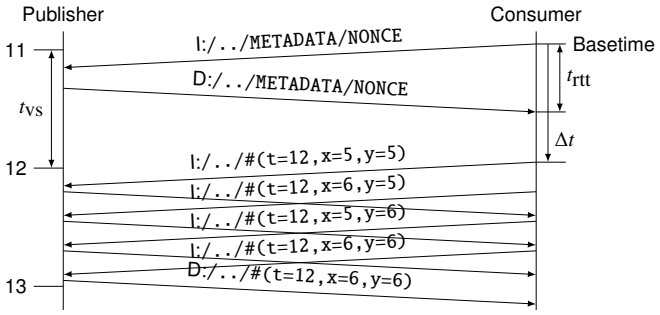

Fig. 5. Message flows between producer and consumer: I: and D: means Interest and Data packet, respectively.

cluding the monitoring system. The consumer is implemented in JavaScript with ndn.js [10] which is an NDN stack in pure JavaScript. The consumer does not have the forwarder, i.e., NFD, but it constructs NDN packets and sends them to the neighbor forwarder via WebSocket. The producer captures equirectangular frames from an omnidirectional camera [14]. When it receives an Interest packet issued by the consumer, it clips the captured frame and encodes it to construct the required Data packet. To avoid encoding tiles that are not required, the producer creates a Data packet after the tile corresponding to the packet is requested. The created Data packets are cached on the intermediate routers; thus, the producer does not need to create the same packet twice. These image processes including video encoding are executed using OpenCV [9].

The transcoder is located at edge nodes. When it receives an Interest packet with a *transcode* request, it sends the Interest packet to get the corresponding high-quality tile and converts it to a low-quality tile. It is important to note that if the processing is completed within the PIT lifetime, the transcoder does not need to be aware of the transaction, i.e., the relationship between the request and response. The ICN router covers complex transaction processing, e.g., duplicate requests. The consumer requests a low-quality tile, when three or four corners of a tile are out of view.

The monitor collects the statistics from the consumer and the edge node for the evaluation and the demonstration [4]. It visualizes the behavior of the system to ascertain the performance.

B. Protocol between Consumer and Producer

Figure 5 shows an example of the message flow between the consumer and the producer. First, a consumer makes a *metadata* request to the producer. This has two objectives. The first one is to get the video streaming parameters, such as video resolution, frame rate, tile size, video segment length and current time sequence number. These acquired parameters are used for playing the movie and for other purposes. The second objective is to measure the RTT (round trip time) between the consumer and the producer. The RTT is used when the consumer decides the timeout period for the Interest packet or the request time sequence number. To avoid the effects of cache and Interest aggregation, the metadata is required with a nonce, which is a random string. This request is sent at regular intervals, e.g., every 5 seconds, in order to update to the latest information. This procedure also helps the producer to ascertain the number of viewers.

The consumer simultaneously requests the tiles needed according to its field-of-view. The video stream is divided into video segments with a constant length, e.g. 500ms, and the consumer estimates the latest video segment number by the following equation:

$$\textit{Video Segment Number} = \frac{\Delta t}{t_{\mathrm{vs}}} + c$$

where Δt is the time that has elapsed since metadata was requested, and t_{vs} and c are respectively the video segment length and the current segment number maintained in the metadata.

TABLE I
EQUIPMENT SPECIFICATIONS

Nodes	
CPU	Intel Core i7 3.0GHz
OS	macOS 10.14.1
Memory	1600MHz DDR3 SDRAM
Forwarder	NFD 0.6.4
Omnidirectional Camera [14]	
Resolution	3840×1920
Frame Rate	29.97 fps

IV. EVALUATION

A. Environment

We evaluated the validity of the system design and the influences of the system parameters by using a simple environment as shown in Fig. 3. Each node is connected with a fixed line which has sufficient bandwidth. Table I shows the equipment specifications. The consumers are emulated on a node, and select their field-of-view randomly. The emulated consumer operates only its pan and tilt, and does not operate its roll or zoom. This is due to the assumption of usage on a VR headset.

The producer provides tiles having two levels of quality, i.e., high-quality and low-quality tiles. A low-quality tile has half the resolution of a high-quality tile, i.e., weight and height divided by $\sqrt{2}$. The captured video frames are encoded into video segments every 500 ms. In order to improve responsiveness, just a shot video segment length is set. A video segment is divided into Data packets with a payload length is 4.4 kbytes, which is half of the maximum NDN packet length. The payload size is a typical value used in sample programs.

B. Tile Size

The tile size greatly affects the performance of our system. Small tiles can represent the user's field-of-view precisely because of their fine-grained representation. However, small tile size lowers the compression ratio, and configuring a size that is too small increases the total data volume needed for transmitting a field-of-view. Figure 6 illustrates the average

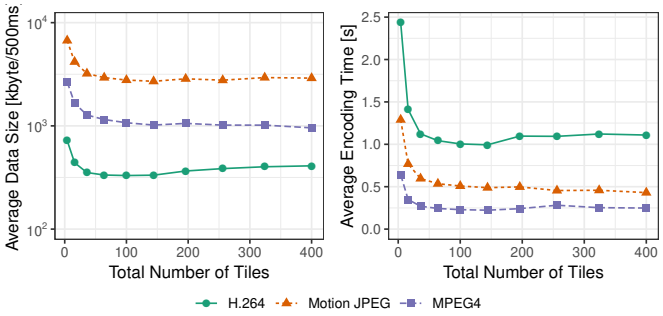

Fig. 6. The average of data amount and encoding time representing a user's field-of-view: The number of tiles is changed from 1 (1×1) to 400 (20×20).

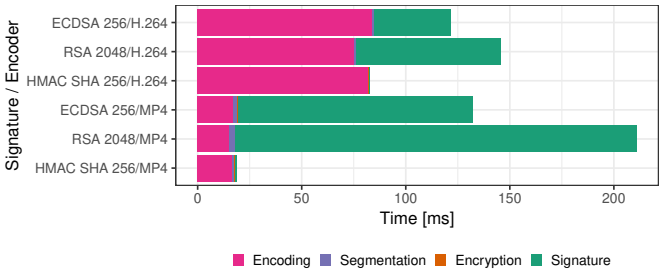

Fig. 7. The average of the processing time to create Data packets representing a tile. The encryption algorithm is AES-128. HMAC uses the same key of the encryption (128bit).

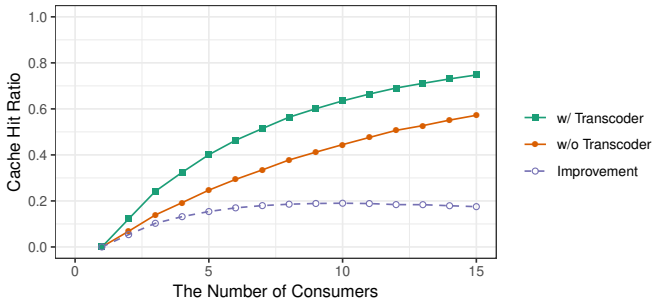

Fig. 8. Cache hit ratio on the edge node. Improvement means the difference between two cache hit ratios.

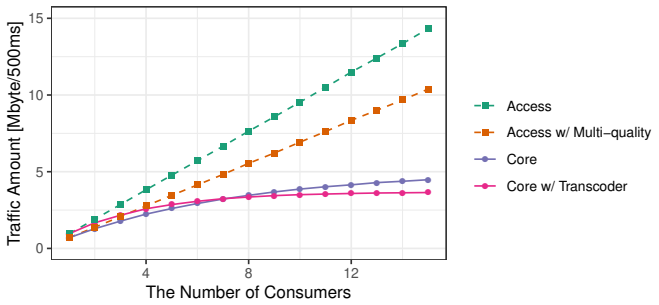

Fig. 9. Traffic volume on the access network between the edge node and consumer, and on the core network between the producer and the edge node.

data size and encoding time for 500 ms (15 frames) representing one field-of-view with various tile sizes and encoders. Where the number of tiles is 100 or more, the reduction in the data size and the encoding time becomes saturated. The tiling technique reduces the data size from 40 to 45% compared with the no tiling case, i.e., where the number of tiles is 1. The following evaluations use 100 (10×10) as the number of tiles.

H.264 makes video data smaller than other methods. But it requires more processing time for encoding. This tradeoff affects the creation of Data packets representing tiles. The next section will evaluate this effect.

C. Data Packet Generation

When the producer receives an Interest packet, it encodes the required tiles, divides into packet segments, encrypts, makes the signature and sends them as Data packets. NDN requires the signature for each Data packet to verify its creator. However, the signature process is computationally heavy, and its processing time cannot be ignored. Figure 7 shows the processing time needed to create Data packets representing a single tile. The processing time for the signature depends on the number of Data packets. H.264 requires more processing time but generates less video data, thus the total processing times of H.264 are less than that of MP4 in the public key-based signature method, i.e., RSA 2048 and ECDSA 256. Specifically, MP4 and H.264 generated an average of 11.6 and 3.8 Data packets per tile, respectively. On the other hand, the symmetric key based signature method, i.e., HMAC, provides a light-weight signature process.

At worst, the system needs to generate all tiles, i.e., 100 tiles, during the video segment length, i.e., 500ms, for live streaming. Since each generation is processed in parallel, it is necessary to complete the process within the following time:

$$s/mp$$

where s, m and p are the video segment length, the number of tiles and the number of parallel processes, respectively. Our system uses "HMAC/MP4" since other methods require more than 10 parallel processing tasks. Section V-C will discuss the reduction of this signature overhead.

D. Traffic Volume

As explained in Section II-B, the transcoder improves the decrease in the cache hit ratio because it reduces the (resolution) heterogeneity of the tiles transferred in the core network. Figure 8 shows the cache hit ratio of the edge node and we see that the transcoder improves the cache hit ratio by up to 19%. However, the transcoder always requires a high-quality tile, even if the consumer is requesting a low-quality one; and high-quality tiles need a greater core network bandwidth if not cached in the edge node. Thus, there is a trade-off between the decrease in the traffic volume due to the increase in cache hit ratio and the increment of traffic volume due to greater requests of high-quality tiles.

Figure 9 shows the traffic volume measured on the edge node. The difference in the traffic volume of the access

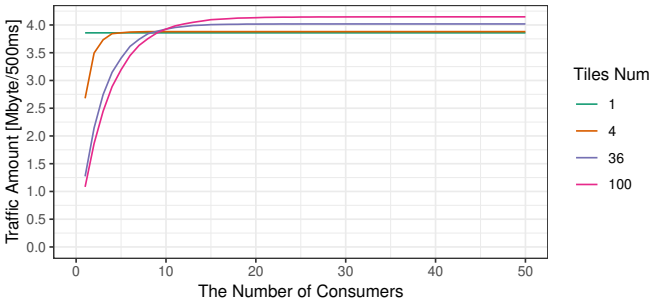

Fig. 10. Simulated traffic volume on the core network between the producer and the edge node. The encoding method is MPEG4.

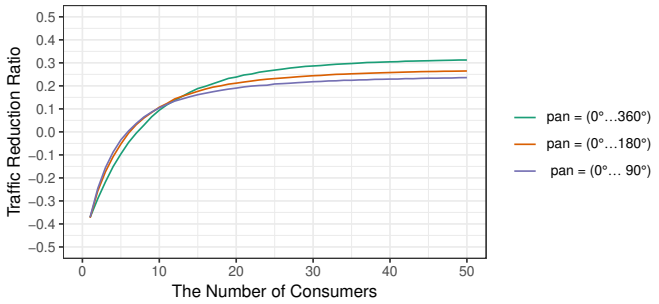

Fig. 11. Traffic reduction ratio by transcoder. A negative value means that the traffic volume has increased by the transcoder.

network, i.e., outgoing traffic from the edge node, illustrates the effect of multi-level quality tile request strategy. It reduces the traffic volume on the access network by about 27%. The traffic volume of the core network, i.e., incoming traffic to the edge node, becomes smaller than that of the access networks due to the effect of caching. On the other hand, the influence of the transcoder on the core network traffic volume is limited. It reduces traffic volume by only 18% at maximum, and conversely increases volume if the number of consumers is small. This is due to the trade-off mentioned above. Even if the effect of reducing the traffic volume on the core network is small, edge transcoding has the merit of offloading producer load limiting the number of requested resolutions. Section V-B will discuss the traffic volume on the core network.

V. Discussion

A. Traffic Reduction by Tiling

In Section IV-B, the tile size that would minimize the traffic volume for each consumer was determined. This optimization is valid, since the access network becomes a bottleneck in the current Internet. However, if the access network has sufficient bandwidth, delivering the same data to all consumers, like many current video distributions systems do, uses the cache most effectively.

Figure 10 shows the relationship between the number of consumers and the simulated traffic volume on the core network for different number of tiles per 360° frame. When the number of tiles is 1, i.e., the producer sends the entire 360° video frame to all consumers, the traffic volume does not depend on the number of consumers. Tiling reduces the traffic volume when the number of consumers is small because they request only a small part of a 360° frame. However, as the number of consumers increases, the traffic volume increases because consumers request the most of a 360° frame and it is more efficient from a compression point of view to use larger tiles. Saturation takes place when all parts of 360° frames are continuously requested. Comparing the case where the number of tiles is 1 and 100, the maximum volume of core network traffic increases by 8%. Thus, it may be more efficient to transmit the whole 360° video frame. However, the coding overhead is only 8%, and the tiling provides more usefulness, such as reducing the traffic volume for a consumer and the multi-level quality tiles.

B. Traffic Reduction by Edge Transcoding

The result presented in Section IV-D showed that the influence of the core network traffic reduction brought ahout by the transcoder is limited. However, it depends on the location of the transcoder. Figure 11 shows the simulated traffic reduction ratio calculated by the following equation where the number of consumers is large:

$$\textit{Traffic Reduction Ratio} = \frac{T - T_{tc}}{T}$$

where T and T_{tc} represents the incoming traffic volume on the edge node without and with a transcoder, respectively. This result confirms that the reduction ratio is more than 30% where the number of consumers exceeds 37. Then the transcoder should be put in a location where is can be accessed by many consumers, e.g., a location close to the root of a tree topology. Additionally, if cache nodes without the transcoder are put near the consumer, the traffic volume of the entire network will be reduced efficiently.

VR content is not uniformly seen in all directions, and many people watch a specific place, such as a music concert stage or a sports stadium. If certain tiles are frequently requested, the cache effect tends to become large. Figure 11 also shows the traffic reduction ratio where the consumer's horizontal angle is resricted, i.e., the restriction of panning. When the number of consumers is small, the reduction rate rises slightly due to the restriction. However, its saturation value is decreasing. The effect of the transcoder is lowered because the cache hit ratio without the transcoder is increasing.

Therefore, if the transcoder is installed in a place where it can receive requests from several consumers, it will reduce the incoming traffic volume by 20 to 30%.

C. Signature Processing

As Section IV-C said, the signature processing has a great influence on the performance of the producer. Marchal [15] evaluated the producer-side performance and also got the result that asymmetric cryptography became a bottleneck. HMAC, which uses a symmetric cryptography, provides a shorter processing time. But it requires that all consumers and

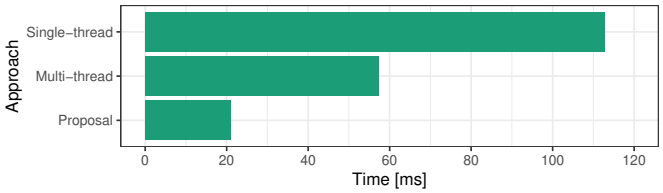

Fig. 12. The average of signature processing time representing a tile. MPEG4 and ECDSA are used as the encoding method and the signing method, respectively.

the producer have the same symmetric key. This means that anyone who is a consumer can sign a Data packet. It is not enough to fulfill the signature's purpose to verify its creator. Kurihara [16] proposed a security model originated on the manifest. The manifest carries hash digests of Data packets, and a consumer verifies packet integrity using the digest. Only the manifest is signed by the asymmetric cryptography to keep the security while reducing the calculation load. However, on our system, not only producer but also transcoder creates the tile dynamically, then it is difficult to prepare the manifest in advance.

We propose a signature scheme inspired by the manifest-based scheme to get the both of integrity and responsiveness. The first Data packet, i.e., the segment number is 0, has hash digests of the all the subsequent Data packets making up the tile in its meta-info field, and is signed using the asymmetric cryptography. The subsequent Data packets are verified their integrity by the digests on the first Data packet. On this method, the first Data packet acts like a manifest.

Figure 12 shows the signature processing time for one tile. The multi-thread signs packets in parallel using 4 threads to reduces the completion time. However the proposal makes it even shorter. Although this result depends on the video data length which is the average of 11.6 Data packets in this evaluation. Since a consumer usually requests some tiles simultaneously, so the merit of parallel processing is obtained by processing individual tiles.

VI. Related Work

Tile-based panoramic video streaming with multi-level quality is a well-known techniques. D'Acunto [17] proposed an MPEG-DASH SRD client to optimize the delivery of zoomable videos, which are affected by the same bandwidth problem as panoramic videos. Feuvre [18] proposed to use H.265 to spatially divide the panoramic video. The aim of these study was to address the high bandwidth requirements of panoramic videos on the access network. They did not discuss the bandwidth on the core network and leveraging in-network resources.

Video distribution is a typical application to leverage ICN features [19]. Morizono [20] proposed *Symbolic Interest* to avoid the bursty requests for the real-time streaming. Detti [21] used peer-to-peer communication to increase the quality of live video playback. These works do not compete with our system, but can be utilized for distribution of tiles.

VII. Conclusion

This paper described the design of a tile-based panoramic live video streaming system to leverage in-network resources, which are key components of the future Internet architecture. For this purpose, it is necessary to design the application based on the required pieces of data. ICN is suitable for utilizing these components due to its data-driven communication scheme. We have a plan to evaluate this video streaming application on a global testbed as the future work.

References

[1] Cisco Systems, Inc., "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021," February 2017.
[2] B. Ahlgren *et al.*, "A Survey of Information-Centric Networking," *IEEE Communications Magazine*, vol. 50, no. 7, 2012.
[3] K. Ueda *et al.*, "Demo: Panoramic Streaming using Named Tiles," in *Proceedings of ACM Conference on Information-Centric Networking*, ser. ICN, September 2017, pp. 204 – 205.
[4] A. T. others, "Demo: Edge transcoding with name-based routing," in *Proceedings of ACM Conference on Information-Centric Networking*, ser. ICN, September 2018.
[5] Y. Sánchez *et al.*, "Compressed Domain Video Processing for Tile based Panoramic Streaming using HEVC," in *Proceedings of IEEE International Conference on Image Processing*, ser. PV, September 2015, pp. 2244–2248.
[6] S. Petrangeli *et al.*, "An HTTP/2-Based Adaptive Streaming Framework for 360° Virtual Reality Videos," in *Proceedings of ACM Multimedia Conference*, ser. MM. ACM, 2017, pp. 306–314.
[7] R. Skupin *et al.*, "HEVC Tile Based Streaming to Head Mounted Displays," in *Proceedings of IEEE Consumer Communications Networking Conference*, ser. CCNC, January 2017, pp. 613–615.
[8] O. Ascigil *et al.*, "A Keyword-based ICN-IoT Platform," in *Proceedings of ACM Conference on Information-Centric Networking*, ser. ICN, 2017, pp. 22–28.
[9] OpenCV team. (2018, October) OpenCV library. [Online]. Available: https://www.opengl.org/
[10] NDN Project. (2018, October) Named Data Networking (NDN) – A Future Internet Archtecture. [Online]. Available: https://named-data.net/
[11] three.js authors. (2018, October) JavaScript 3D Library. [Online]. Available: https://threejs.org/
[12] InfluxData, Inc. (2018, October) Time Series Database Monitoring & Analytics. [Online]. Available: https://www.influxdata.com/
[13] Grafana Labs. (2018, October) The Open Platform for Analytics and Monitoring. [Online]. Available: https://grafana.com/
[14] Ricoh. (2017, April) RICOH THETA. [Online]. Available: https://theta360.com/
[15] X. Marchal *et al.*, "Server-side Performance Evaluation of NDN," in *Proceedings of ACM Conference on Information-Centric Networking*, ser. ICN, 2016, pp. 148–153.
[16] J. Kurihara *et al.*, "An Encryption-based Access Control Framework for Content-Centric Networking," in *Proceedings of IFIP Networking Conference*, ser. IFIP Networking, May 2015, pp. 1–9.
[17] L. D'Acunto *et al.*, "Using MPEG DASH SRD for Zoomable and Navigable Video," in *Proceedings of International Conference on Multimedia Systems*, ser. MMSys. ACM, 2016, pp. 34:1–34:4.
[18] J. Le Feuvre and C. Concolato, "Tiled-based Adaptive Streaming Using MPEG-DASH," in *Proceedings of International Conference on Multimedia Systems*, ser. MMSys. ACM, 2016, pp. 41:1–41:3.
[19] C. Westphal *et al.*, "Adaptive Video Streaming over Information-Centric Networking (ICN)," Internet Requests for Comments, RFC Editor, RFC 7933, August 2016.
[20] K. Matsuzono and H. Asaeda, "NMRTS: content name-based mobile real-time streaming," *IEEE Communications Magazine*, vol. 54, no. 8, pp. 92–98, August 2016.
[21] A. Detti *et al.*, "Peer-to-peer live adaptive video streaming for information centric cellular networks," in *Proceedings of International Symposium on Personal, Indoor, and Mobile Radio Communications*, ser. PIMRC. IEEE, September 2013, pp. 3583–3588.