Third-Party Preissuance Submission regarding patent application US201615370840:
"Mixed boolean-token ans coefficient coding” by Alexander Jay Converse, Google Inc.

Dear Sir/Madam,

I am the author of the Asymmetric Numeral Systems (ANS) family of entropy coders (e.g. tANS, rANS, rABS, uABS variants), whose natural usage is covered in the patent application of concern. Due to much better performance, since 2014 ANS is becoming practically the default way of entropy coding, replacing standard methods like Huffman and arithmetic coding. ANS is currently used in more than a dozen of data compressors, including ones from well known companies like Apple, Facebook and Google.

A crucial reason for its popularity was my will and actions to make it public domain, protect its use from becoming a legal minefield. As patents have already prevented wide adoption of arithmetic coding for many decades, I wanted to do all I could to make public the required basic concepts, modifications and applications by presenting them in scientific papers, reference implementations, and multiple public discussion forums. Many other data compression specialists have joined this initiative to prevent monopolization of natural new possibilities coming with the ANS coders, by making public (as open-source software) their implementations and written reflections in forms of scientific articles, statements in public forums and blog posts, some of which can be accessed through [1].

However, as this technology has turned out extremely valuable, there have emerged attempts to monopolize its basic use and applications. A previous patent application (US201615041228) presented claims regarding natural combination of ANS and Markov modelling, even though it was explicitly mentioned in my earlier articles, as well as being used before in a known in the society open-source implementation developed by other person, in that case, a direct competitor of the applicant (both work in DNA compression).

The Google patent application of concern is the second attempt I am aware of. I have discovered it accidentally after its publication, which was very surprising for me as I was helping the Google video compression group exactly on this topic since January 2014 through public discussion forum of this group [2] and emails. As I will demonstrate below, the claimed innovation is an obvious modification of a textbook method to exploit the advantages coming with ANS. Additionally, I have directly suggested it them in our 2014 communication. However, they did not consult this patent application with me, did not inform me about the application, and did not include me as a co-author, which raises serious ethical concerns regarding this unjustified attempt of monopolization.

The content of this patent application is a direct natural modification of a textbook way for encoding transform coefficients that represent image blocks in video/image compression. This approach is well known, for example from the popular CABAC codec used in H.264 video compressor and proposed around 2003 [3]. In addition to binary symbols (like flags marking events), image and video compression requires to encode symbols from large alphabets (e.g., DCT or Fourier transform coefficients). Due to the lack of methods for efficient direct processing of large-alphabet symbols in 2003, CABAC binarizes them (i.e., transform coefficients in the application at hand). Hence, it splits a single choice among a large number of possibilities, into a few binary choices. The main advantage provided by ANS is inexpensive direct handling of large alphabet symbols – allowing to replace costly "binarization with a few binary steps" with a simple single step for a large alphabet.

In many image and video compressors, transform coefficients are coded in a sequence that puts important and large coefficients first, and leaves long clusters of unimportant, zeroed coefficients at the trailing end. Such an approach allows to encode only the important coefficients, and just signal that after some point all the remaining coefficients are zero. This is achieved by an EOB marker in JPEG images and a last significant coefficient flag in CABAC (last_significant_coeff_flag[]).

In order to implement this functionality with ANS, the concerned patent application proposes to switch between a binary alphabet (used for encoding the flag) and a large alphabet (used for encoding the coefficients). The mechanism requires separate coders for each mode – conventional ANS for the coefficients (referred as "token" in title of this patent), and an ABS coder (a binary variant of ANS, referred to as "boolean" in title of this patent) for the flag.

The above mechanism, as well as its application to video compression were explicitly mentioned in my scientific paper [4] (cited in this patent application) from January 2014. I have written: "*While maintaining the state space (I), we can freely* ***change between*** *different probability distributions, alphabets or even* ***ABS/ANS variants****. For example for complex tasks like* ***video compression*** *(…)*"

Moreover, further suggestions can be found in my posts in the ANS-focused thread [2] on a discussion forum of the Google's image and video compression group, in which the author of this patent application also participates (as Alex Converse). Here are selected fragments from my early statements on this ANS-focused thread:

- 7th January 2014 (my initial post): "*I would like to propose a discussion about the possibility of applying it in* ***video compression*** *like VP9 (…)*",
- 7th January 2014: "*I imagine that video compression is lots of binary adaptive choices - where we should work bit by bit, but there is also lots of encoding of DCT* ***coefficients*** *as large independent numbers, just a few different ranges/contexts (…) we can freely* ***switch*** *between tables symbol by symbol*",
- 8th January 2014 (avoiding binarization): "*The income of encoding e.g. 256 level quantization DCT coefficient, would be* ***instead of 8*** *slower (renormalization) AC steps,* ***a single cheap step of ANS*** *– being about 10 times faster, what for hardware implementation means much lower frequencies and energy usage (smartphones…).*",
- 26th July 2014 (use of rANS variant which is in the final implementation): "*So you are probably using 256 size alphabet range coder for a lot of fixed probabilities - rANS is its many times faster direct alternative*"
- 27th May 2015 (information about LZNA compressor which also switches between ABS and ANS): "*Finally what video compression needs: probably the first adaptive rANS compressor - LZNA - promised to usually provide better compression than LZMA, while having a few times faster decompression(…)*".

The concerned patent application also briefly introduces well-known basic techniques of ANS (like state renormalization, handling probabilities), used by dozens of people in various public implementations [1]. While the implementation I have helped them with was for a specific variant of ANS (rANS variant to be exact), this patent application is written in a more general way to restrict free use also of other ANS variants (especially tANS).

To further demonstrate the obvious character of the claimed inventions, I'll take the liberty to briefly discuss two connected earlier ANS-based compressors. For example GST image compressor also uses ANS to encode transform coefficients. The relevant scientific article was published 6th November 2016 [5], but the Github repository of this open-source project allows to trace its first use of ANS to 17th November 2015 [6]. An example for earlier use of mixing binary and multisymbol variants (ABS and ANS – the same ones as discussed in the patent application) can be found in LZNA compressor of RAD Tool Games. It was announced 5th October 2015 [7] and later its implementation [8] has shown that it switches between 3 different implementations for different sizes of alphabets: binary, 3-bit and 4-bit, what can be seen as an extension of what is used in the concerning solution of Google (mixing binary and 4-bit variants).

Google did neither respect my will to prevent legal restrictions for using ANS, nor did consult with me or inform me about this patent application, whose claimed innovations can be found, among others, in my well documented work for and with them.

Despite dubious innovation claims, this application can be seen as a legal risk for both the existing ANS-based image compressors (like GST) and for other parties considering ANS for future image and video compressors.

Therefore, I am requesting the rejection of this application, eventually I kindly ask to remove direct referring to ANS in this patent applications.

Sincerely yours,

Jarosław Duda, PhD

Jagiellonian University, Cracow, Poland.

Email: jaroslaw.duda@uj.edu.pl

Sources:

[1] https://encode.ru/threads/2078-List-of-Asymmetric-Numeral-Systems-implementations

[2] https://groups.google.com/a/webmproject.org/forum/#!topic/codec-devel/idezdUoV1yY

[3] Marpe, Detlev, Heiko Schwarz, and Thomas Wiegand. "Context-based adaptive binary arithmetic coding in the H. 264/AVC video compression standard." *IEEE Transactions on circuits and systems for video technology* 13.7 (2003), Available: http://iphome.hhi.de/marpe/download/cabac_ieee03.pdf

[4] J. Duda, arXiv:2540 (2014), Available: https://arxiv.org/pdf/1311.2540v2.pdf

[5] P. Krajcevski, S. Pratapa, D. Manocha, "GST: GPU-decodable supercompressed textures", ACM Transactions on Graphics (2016), Available: http://dl.acm.org/citation.cfm?id=2982439

[6] https://github.com/GammaUNC/GST/commit/9c8a822e9820b3d1abd1b628041e53f0acaf8f59

[7] http://cbloomrants.blogspot.de/2015/05/05-09-15-oodle-lzna.html

[8] https://github.com/powzix/kraken/blob/master/lzna.cpp