Earth+: On-Board Satellite Imagery Compression Leveraging Historical Earth Observations

Kuntai Du
University of Chicago
Chicago, IL, United States
kuntai@uchicago.edu

Yihua Cheng
University of Chicago
Chicago, IL, United States
yihua98@uchicago.edu

Peder Olsen
Microsoft Research
Redmond, WA, United States
Peder.Olsen@microsoft.com

Shadi Noghabi
Microsoft Research
Redmond, WA, United States
shadi.noghabi@gmail.com

Junchen Jiang
University of Chicago
Chicago, IL, United States
junchenj@uchicago.edu

Abstract

Due to limited downlink (satellite-to-ground) capacity, over 90% of the images captured by the earth-observation satellites are not downloaded to the ground. To overcome the downlink limitation, we present Earth+, a new on-board satellite imagery compression system that identifies and downloads only *changed areas* in each image compared to *latest* on-board reference images of the same location. The key of Earth+ is that it obtains latest on-board reference images by letting the ground stations *upload* images recently captured by *all* satellites in the constellation. To our best knowledge, Earth+ is the first system that leverages images across an entire satellite constellation to enable *more images* to be downloaded to the ground (by better satellite imagery compression). Our evaluation shows that to download images of the same area, Earth+ can reduce the downlink usage by 3.3× compared to state-of-the-art on-board image compression techniques without sacrificing imagery quality or using more resources (downlink, computation or storage).

CCS Concepts: **• Computing methodologies → Image compression**; **• Information systems → Geographic information systems**; *Sensor networks*; **• Computer systems organization →** *Embedded software*.

Keywords: satellite imagery compression, reference image, downlink optimization, earth observation satellites

ACM Reference Format:
Kuntai Du, Yihua Cheng, Peder Olsen, Shadi Noghabi, and Junchen Jiang. 2025. Earth+: On-Board Satellite Imagery Compression Leveraging Historical Earth Observations. In *Proceedings of the 30th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 1 (ASPLOS '25), March 30–April 3, 2025, Rotterdam, Netherlands.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3669940.3707222

ASPLOS '25, March 30–April 3, 2025, Rotterdam, Netherlands

ACM ISBN 979-8-4007-0698-1/25/03.
https://doi.org/10.1145/3669940.3707222

1 Introduction

Fresh and high-quality satellite imagery is key to many applications, from digital agriculture [29, 56, 72, 73], environmental monitoring [6, 46, 68, 84, 85], to automatic road detection [31, 60, 86], and many more. As a result, large constellations of Low-Earth-Orbit (LEO) earth observation satellites have been deployed [48, 74, 82] to capture high-quality imagery for any location multiple times a day [48, 74, 82].

However, most satellite imagery data captured by these satellites are currently *not* received on the ground due to the limited *downlink* (satellite-to-ground) capacity. According to a recent estimate, only 2% of the total image data observed by each satellite can be downloaded to the ground [48]. Some mission-specific satellites handle the downlink-capacity limitation by filtering images onboard the satellite [49, 82] to focus only on mission-specific areas prepaid by the customer. However, this approach is not sufficient for *general-purpose* satellite constellations (*e.g.,* Sentinel-2 [51], Doves [74]), whose goal is to capture and download satellite imagery over wide geographical regions to serve more applications.

This paper aims to improve *onboard compression* for satellite imagery.[1]. We are inspired by the observation that the terrestrial content changes slowly between two consecutive satellite visits at the same location [78, 88]. Thus, to compress a new image, we can compare it with a recent image of the same region, called a *reference* image, to detect the geographic *tiles* (defined in §3) within the region that has changed and then only compress and download the changed tiles. Our measurement on Planet dataset [74] shows that without the interference of clouds, only 20% of the tiles in

* This work was conducted while Peder Olsen and Shadi Noghabi were affiliated with Microsoft Research.
[1]Better imagery compression can also improve filtering-based solutions like [48, 82].

each image have changed in the previous five days on average, which ideally can save downlink usage by up-to 5× (§3).

Yet, realizing the *reference-based* encoding for onboard imagery compression can be challenging because the reference image should be as *fresh* and contain as little *cloud* as possible (§3). Typically, the last cloud-free image captured by the same satellite [74] can be over 50 days old on average (§3). With such a large time gap, the reference image and the new image may have substantial differences (more than 50% of the tiles will have significant changes as shown in §3), making reference-based encoding less effective.

We present Earth+, a *constellation-wide* reference-based encoding system, where the reference images can be selected from historical images of *any satellites* in the constellation. By broadening the set of potential reference images, Earth+ increases the probability of obtaining fresh and cloud-free reference images. For example, with images from an entire constellation [74], cloud-free images can be obtained every 4.21 days on average, instead of every 50 days with one satellite (§4.1).

Earth+ then leverages the existing *uplinks* (ground-to-satellite) to *upload* reference images selected from the whole constellation to the target satellite, as illustrated in Figure 1. (§4.2 will discuss why Earth+ does not leverage inter-satellite links instead.) The key challenge of this design is to handle limited uplink capacity of existing earth observation satellites (*e.g.,* 250kbps [50]).

We present two techniques (§4.3) to reduce the uplink usage of Earth+ without sacrificing the savings on the downlink.

First, Earth+ uploads reference images at a low resolution while still allowing the satellites to detect the most changed tiles (§4.3). The rationale is that low-resolution images are sufficient to decide which tiles have changed, which is easier than quantifying how much each pixel in the tile has changed.

Second, Earth+ does not need to store those unchanged tiles when capturing new imagery, which frees up the storage space. We utilize this freed storage space to cache reference images locally on-board, which allows Earth+ to further reduce the uplink usage by only uploading tiles that have changed relative to the on-board cached reference images.

Besides the two aforementioned techniques, our implementation of Earth+ (§5) also entails techniques to handle satellite-specific issues, including cloud detection, on-board computation constraints, handling different bands of satellite imagery, and bandwidth variations.

To put Earth+'s contribution into perspective, the idea of sharing imagery across satellites in the constellation is not new (*e.g.,* multipath satellite imagery delivery [7, 66]). Earth+, however, is the first that leverages constellation-wide imagery sharing to enable *more images* to be downloaded to the ground (by better satellite imagery compression).

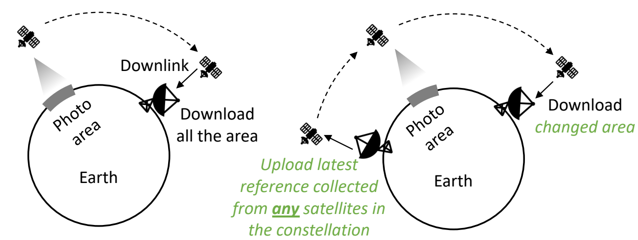


Figure 1. *Contrasting Earth+ with traditional satellite imagery compression. (a) The traditional approach compresses images by satellites using their local onboard information. (b) Earth+'s reference-based encoding uses reference images from any satellites in the constellation and uploads the reference images to the satellite to identify and download only the changed areas.*

We evaluate Earth+ on *real-world* satellite specifications (uplink and storage capacities) of the Doves constellation [25] from Planet Labs. We test Earth+'s compression efficiency on two datasets. The first dataset is collected from Sentinel-2 dataset [51], with 3.6 TB data covering 110 thousand km^2 from Washington State. We use this dataset to test Earth+ under a wide range of contents (*e.g.,* mountains, forests, and cities), seasons, and under multiple imagery bands (13 bands in total). Since Sentinel-2 only contains two satellites, we further test Earth+'s performance using the Planet dataset [74], from which we obtain images from 40 satellites for one sampled location (due to the download limit) of 64 km^2 in the U.S. for three months. Our evaluation shows that:

- Compared to the state-of-the-art onboard compression schemes, Earth+ reduces the downlink bandwidth usage by 1.3-3.3× without hurting the imagery quality on all bands. This can reduce the reaction delays of ground applications (*e.g.,* forest-fire alerts) by upto 3×.
- These improvements are achieved *without* using more uplink bandwidth than currently available or more compute or storage resources than the baselines.
- With more satellites in a constellation, Earth+ can further reduce the amount of downlink bandwidth usage.

That said, Earth+'s reference-based encoding is not a good fit for applications that require lossless satellite imagery (§8).

2 Motivation

We start with the background on satellite imagery and earth observation satellite constellations.

2.1 Background

Many applications can benefit from frequently updated (*e.g.,* daily) and high-resolution satellite imagery. For example, precision agriculture ideally needs daily access to satellite imagery with each pixel corresponding to a 5m × 5m area on Earth [14, 30] to help timely decisions on the distribution

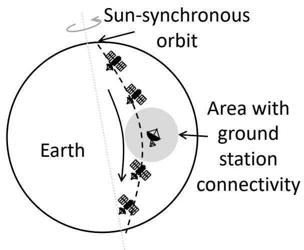


Figure 2. *Illustration of a LEO satellite constellation: the satellites follow a sun-synchronous orbit, and the ground station downloads imagery data from a satellite when the satellite passes through it.*

of fertilizers, pesticides, and water. Also, wildfire monitoring requires the imagery to be updated frequently with sufficient resolution to promptly detect and respond to fire outbreaks, mitigating potential damage [82].

To provide fresh, high-resolution satellite imagery, many LEO satellites (*e.g.,* >100 satellites [74]) are deployed to form satellite constellations. Figure 2 shows an illustrative example of a LEO satellite constellation, where multiple satellites are located in a sun-synchronous orbit[2] and these satellites can potentially stream data to the ground when they are close enough to one of the ground stations (we only plot one ground station in the figure for simplicity).

We characterize two features of such LEO satellite constellations:

- *High-resolution imagery*: LEO satellites are close to the ground (due to their low earth orbits) and can capture imagery with low ground-sampling distance (GSD for short, lower GSD means higher resolution).
- *Frequent revisit*: With a large number of satellites, any location on the earth's surface will be frequently revisited (*e.g.,* daily [74]), while a single satellite can only revisit one location once every ten days [36].

Note that in the following text, we denote *the ground* as the ground stations that the constellation can potentially contact and the computation and networking infrastructures around these ground stations.

2.2 Downlink bottleneck and our objective

Downlink capacity gap: Despite more images being captured by the satellites, only a small fraction of data are downloaded to the ground due to the limited capacity of the downlink (satellite-to-ground) [48, 66, 82]. Specifically, we refer to *downlink bandwidth* as the average download speed from satellites to the ground during each ground contact. The exact gap between the downlink capacity and the imagery data varies with the constellation, and a recent study shows only about 2% of the images captured by satellites are actually downloaded to the ground [48].

Further, the downlink demand is constantly growing, with higher resolution (*e.g.,* a GSD of 0.5m [32]) and more bands found to be useful (*e.g.,* vegetarian red edge band and water vapor band [33]). In contrast, the downlink grows slowly due to the long deployment cycle of satellites. These trends suggest that the gap between the demand for downlink bandwidth and its actual capacity will likely persist if not increase.

Optimization objective: We aim to address the downlink bottleneck of satellite constellations by better *satellite image compression.* More specifically, we aim to use much less bandwidth to download the same amount of satellite imagery, measured in the number of photoed locations and frequencies, without compromising image quality. To measure the quality of the downloaded images, we use Peak Signal-to-Noise Ratio (PSNR for short), which aligns with satellite imagery compression literature [52, 55, 57, 80].

On-board constraints: While optimizing for the image quality and reducing the downlink consumption, we stick to real-world on-board storage, computation, and uplink constraints. We describe the real-world satellite specification that we used for our evaluation in §6.

2.3 Existing solutions

There are several approaches to addressing the downlink bandwidth bottleneck.

Upgrading infrastructures: The first is to physically increase the total downlink capacity of the satellite constellation by upgrading the infrastructure (*e.g.,* building more ground stations [74] or adding more satellites [48, 51, 74]). The costs of such infrastructure changes can be prohibitive, and they can be slow. For example, it takes tens of millions of dollars to build and send just one single satellite [11].

On-board filtering: An alternative is to filter the imagery onboard the satellite [48, 49, 82]. For the mission-specific constellations that focus on specific regions, this approach can filter out most of the imagery. For instance, the Biomass mission targets forest areas to monitor forest coverage changes [3], while the IceBridge mission observes polar ice to gauge climate change impacts [1]. However, they must exclude data useful for other applications. For example, the Biomass mission omits about 91% of the Earth's surface [10, 13], such as city areas (which are useful for smart city applications) and agriculture areas (useful for digital agriculture).

In-space application processing: Instead of downloading the imagery to the ground, a wide range of systems process the application onboard the satellite and stream the application results back to the ground [2, 15, 16]. However, this approach cannot support many applications due to limited on-board compute, while downloading imagery to the ground allows all applications to perform analytics based on downloaded imagery.

[2] A sun-synchronous orbit ensures that each location on Earth is revisited once per day, at approximately the same time local time, allowing the constellation to capture images with similar illumination condition once per day.

Inter-satellite link for multi-path imagery delivery: Boosted by coherent optical communication [59, 77, 87], the inter-satellite link capacity is quickly growing and allows multipath satellite imagery delivery that can significantly reduce imagery delivery latency [61, 66]. However, this approach does not increase the total downlink capacity of the satellite constellation, or reduce the total amount of imagery data that need to be downloaded, so it is still bottlenecked by limited downlink capacity.

On-board satellite imagery compression: This work focuses on onboard imagery compression, which is complementary to the first three approaches. Existing solutions include augmenting single-image codecs [37–39, 58, 67, 75, 76, 93] and developing more expensive neural-based codecs such as autoencoders [40, 41, 47, 95, 96]. However, these techniques focus on compressing single imagery from a single satellite, so they fall short in leveraging the *redundancies between* images for higher compression efficiency.

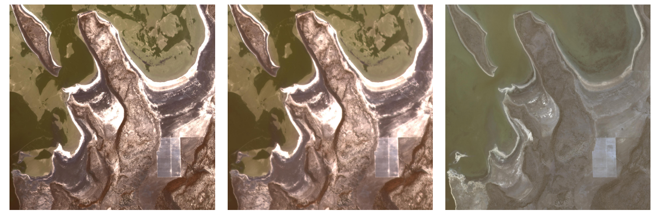
3 Reference-based encoding

Next, we introduce reference-based encoding, a seemingly promising idea that leverages a reference image to pinpoint and download only regions that have recently changed. As we will see, directly applying this approach to a satellite does not work well as images locally available to each satellite may not be recent enough or contain too much cloud to realize the benefit of reference-based encoding.

Background on reference-based encoding: Reference-based encoding is commonly used to compress a sequence of images whose content changes slowly and gradually with respect to time[42, 71, 78, 81, 88, 89], such as video streams. Existing reference-based encoding systems (*e.g.,* video codecs [42, 71, 81, 89]) typically select some of the images as the *reference* and encode the remaining images by encoding their difference concerning the reference images. As existing codecs encode the images at the granularity of *tiles* (a tile is a block of pixels, where we use a 64×64 pixel block as a tile by default), and the difference is separately calculated per tile.

Since the satellite imagery captured for the same location also changes slowly over time (as shown in prior work [78, 88]), there is some recent work to apply reference-based encoding in onboard satellite imagery compression [78, 88]. Given a new image, it compares the image with a reference image of the location from the past and pinpoints the changed tiles with a pixel difference greater than the threshold compared to the reference. It then encodes those changed tiles and downloads the tiles in their entirety.[3] Our work follows this approach when encoding changed tiles (§5).

[3]Unlike conventional video frame encoding, the changed tiles are downloaded in their original pixel values rather than the pixel differences between the new image and the reference. This is mainly because the reference images are downsampled due to limited onboard storage, thus encoding the tile itself and the difference concerning its reference requiring a similar amount of bits.



(a) Captured image (Day 30) **(b)** Fresh reference (Day 27) **(c)** Older reference (Day 1)

Figure 3. *An example illustrating why reference images need to be fresh: If we use an image from Day 27 as a reference to encode the image from Day 30, most areas do not differ from the reference and do not need to be downloaded. In contrast, using the Day 1 image as a reference, most areas differ significantly (due to snow between Day 1 and Day 27) and need to be downloaded. Image © 2023 Planet Labs PBC.*

Reference images need to be *fresh*: While reference-based encoding seems to be a good fit for imagery compression, it is only effective if the *age* of the reference image—the time gap between the reference image and the currently observed image—is as low as possible. Reference image with high age leads to more changed areas in the currently observed image, which must be downloaded to the ground. Figure 3 provides an illustrative example, where the amount of changes need to be downloaded at Day 30, if using high-age reference images from (Day 1), will be much more compared to using low-age reference image (Day 27). To make it more concrete, we use three months of cloud-free (explained shortly) images from the Planet dataset [51] on one randomly sampled location in the U.S. Here, we say a tile has changed if it has an average pixel differences greater than 0.01 after aligning the illumination (§5).[4] Figure 4 shows a steady increase in the percentage of changed areas with the age of the reference image: *the percentage of changed tiles will increase by 3× if increasing the age of the reference image from 10 days to 50 days.*

Reference images should be *cloud-free*: If some tiles in the reference image are covered by clouds, they are not useful as a reference to detect changes. As a consequence, the corresponding tiles in the current image can only be deemed as changed and downloaded to the ground. This greatly compromises the benefit of reference-based encoding.

Why reference-based encoding is challenging? In practice, however, there may not always exist a reference in the satellite's history images that is *both* fresh and covered by little cloud. For example, existing work [78, 88] stores a fixed reference image on-board, which will get older over time

[4]The pixel differences are computed after we normalize pixel values to [0,1]. The threshold of 0.01 means that those areas deemed "unchanged" would be above 40, which is very high in satellite imagery compression literature [43] and provides almost the same results as the uncompressed image in applications like satellite imagery classification [62].

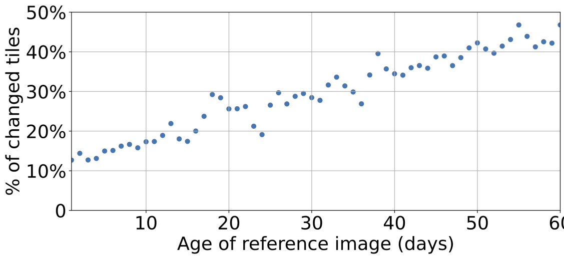


Figure 4. *More changes need to be downloaded when the age of the reference image gets larger.*

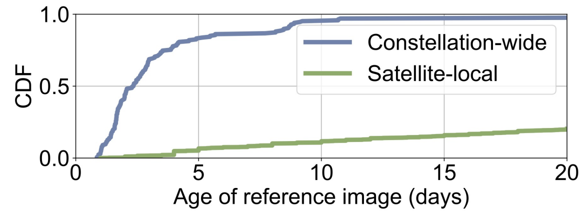


Figure 5. *Measuring the age of reference images under two strategies: updating the reference using historical images locally captured by the satellite ("Satellite-local") and updating the reference using images from any satellites in the constellation ("Constellation-wide"). It shows that the constellation-wide approach can reduce the average age of the reference image* ***from 51 days to 4.2 days****, a* ***12***× *reduction.*

and make most of the areas being counted as changed and downloaded to the ground, negating the benefit of reference-based encoding. Moreover, *even if* a satellite *were* able to choose the reference image from all of its historical images, the most recent reference image with less than 1% cloud coverage would still be tens of days old. For instance, Figure 5 shows the age distribution of the closest reference images that are covered by less than 1% cloud if the satellite chooses the reference image by itself (*i.e.,* the "Satellite-local" curve in the figure). We note that the age of the most recent cloud-free reference image is 51 days on average. The reason for the high ages of recent cloud-free images is two-fold:

- A single satellite revisits the same location at a low frequency (once every 10-15 days [36]). This is because LEO satellites can only capture a small area on Earth at a time (since their size is small [82] and they are close to Earth), necessitating extended periods to complete a full scan of the Earth before revisiting the same locations.
- Since, on average, 2/3 of the earth is covered by clouds [12], so even if the most recent image of the same location is ten days old, it may likely be (partly) covered by cloud and are not ideal choice for reference images.[5]

4 Earth+: Constellation-wide Reference-based encoding

To improve onboard satellite imagery compression, we present Earth+, a reference-based encoding system that obtains fresh and cloud-free reference images from images captured by any satellites in the ***whole constellation***, rather than the history images of the same satellite. This section introduces the idea of constellation-wide reference sharing (§4.1) and an overview of Earth+ (§4.2). We then present the design of Earth+ that makes constellation-wide reference-based encoding practical (§4.3).

[5]Compared to picking a complete image as the reference, choosing a different reference for each tile may lower the age of the reference, but in practice this reduction is marginal because when the cloud is present, it often covers most of an image and the remaining content, if any, will be influenced by its shadow, making change detection difficult.

4.1 *Constellation-wide* reference selection

Compared to the prior work, which only refers to local images observed by the same satellite, Earth+ *augments* the set of reference images that reference-based encoding can choose from and thus potentially reduces the age of reference images, leading to fewer changes to be downloaded to the ground.

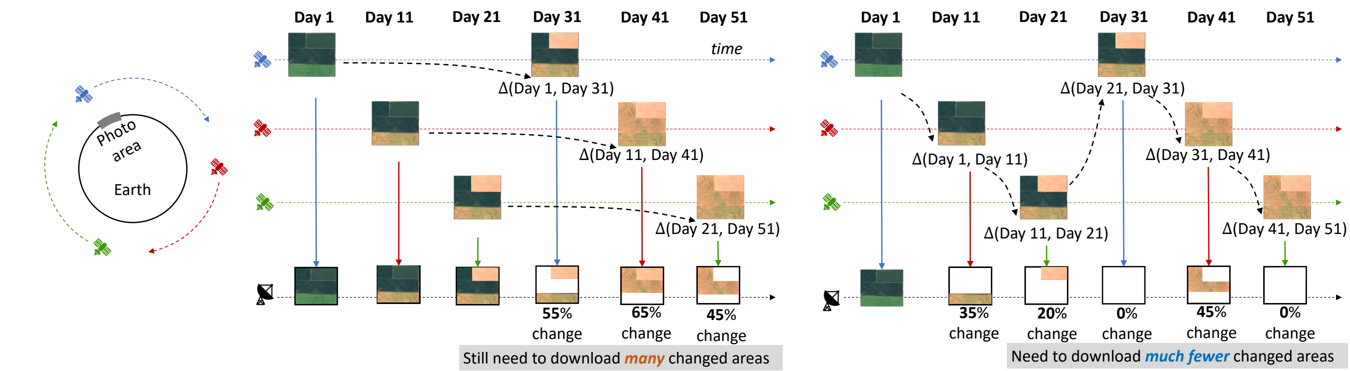
To illustrate the benefits and challenges of Earth+, we contrast two designs.

- *Satellite-local reference:* Pick the latest cloud-free image observed by the same satellite as the reference image.
- *Constellation-wide reference:* Pick the latest cloud-free image observed by ***any satellite in the whole constellation*** as the reference image.

Note that the latter is not practical because it needs a large amount of bandwidth to share the reference images, a challenge we will tackle soon in §4.3.

Figure 6 gives an illustrative example of this contrast with a constellation of three satellites (in different colors). The goal is to compress images taken by these satellites for the same location. To simplify the discussion, all images in this example are cloud-free. Each satellite takes a cloud-free image every 30 days, so the satellite-local reference (Figure 6(b)) will be 30 days old. Consequently, in the last three images (Day 31, 41, and 51), 45%-65% of tiles are deemed as changed and need to be downloaded.

In contrast, with constellation-wide reference Figure 6(c)), since the reference image can be from any satellite, the freshest reference is only ten days old rather than 30 days. As a result, two of the three last images do not have any changed tiles and one has only 45% changed tiles, *i.e.,* only 15% are changed tiles on average. In short, the ability to pick reference images from any satellite in the constellation reduces the age of reference images by 3× (30 days to 10 days) compared to the satellite-local design, and this reduces the changed tiles to download by 3.6× (55% area to 15% area).



(a) A three-satellite constellation photographing one area

(b) Satellite-local: Each satellite uses own history images as reference.

(c) Constellation-wide: Satellites can use recent images from the whole constellation as reference

Figure 6. *Contrasting two reference image uploading strategies: (a) using satellite-local images as reference images and (b) uploading reference images using uplink. By uploading images using uplink, one can reduce the average age of the reference image by 3× (from 30 days to 10 days) and the downloaded areas by 3.6× (55% to 15%) in this example.*

4.2 Earth+ workflow

Earth+ is a concrete design of constellation-wide reference-based encoding. It answers two basic questions: (1) which reference images should be shared between different satellites, and (2) how to share these reference images using the existing infrastructure.

To answer the first question, Earth+ reuses the images downloaded to the ground from all satellites and *selectively uploads* these images as reference images to the satellites. Figure 1(b) illustrates this workflow.

- During **previous** ground contact, the ground station **uploads** latest cloud-free images (that can come from *any* satellite in the constellation) as reference images for the locations that the satellite will fly by before the next ground contact[6].
- When passing over a location, the satellite captures the imagery, removes clouds, detects changes using the reference images, and encodes the changes.
- During the next ground contact, the satellite downloads the encoded changes to the ground.

Compared to the workflow of traditional satellite imagery processing pipelines, which capture images and download them to the ground (as depicted in Figure 1(a)), Earth+**uploads** the reference images from the ground to the satellite. We rely on ground stations as an "overlay" point to share images downloaded from each satellite with other satellites. The rationale is two-fold. First, the ground stations can access any historical image observed by the whole constellation, allowing Earth+ to select reference images constellation-wide. Second, the ground station has sufficient computing resources to more accurately detect clouds and upload only cloud-free images to satellites as the reference (§3).

[6]It is feasible to upload reference images for geographical locations that the satellite will visit in the future, as these locations can be accurately predicted by, for example, Two Line Element data available in Celestrak [4].

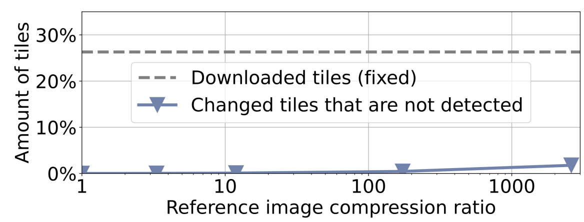


Figure 7. *A measurement based on the Planet dataset showing how many changed tiles that are undetected given different reference image compression ratios, while fixing the total number of tiles being downloaded. We highlight that only 1.7% of the tiles failed to be detected as changed when compressing the reference images by* ***2601****×.*

A seemingly promising alternative to enable constellation-wide reference is to let satellites share data via inter-satellite links (ISL). Earth+ does not use ISL because it is currently not available for earth observation satellites [82]. Further, the scale of existing earth observation constellations (less than 200 satellites) is insufficient to guarantee a stable ISL connection between any two satellites, as providing such a guarantee typically requires thousands of satellites (*e.g.*, Starlink [69]).

4.3 Tackling limited uplink bandwidth

However, using the uplink to upload reference images to the satellites is not without challenges—the uplink has limited bandwidth (*e.g.*, only 250 Kbps in DOVEs constellation [50]). Earth+ tackles this challenge with three ideas. Put together, they allow enough reference images to be sent to the satellites under the limited uplink bandwidth while allowing Earth+ to realize sizable downlink savings.

Downsampling reference images: Earth+ compresses reference images by downsampling (*i.e.*, lowering resolution) and detecting changed tiles at a lower resolution. For example, if the original image is 4000x4000 and the reference image is downsampled to 500x500, the satellite will also downsample the captured image to 500x500 before calculating pixel differences and detecting changes. We then mark the tiles with average pixel difference over a threshold θ (see §5 for details on how to pick θ) as changed tiles and only encode and download these changed tiles.

Detecting changes with downsampled images is less accurate than with full-resolution images. However, we notice that it mainly triggers *false negatives* (*i.e.,* changed tiles might be mis-detected as unchanged). This is because the downsampling essentially averages out the pixel changes in a tile, so the amount of changes are lower compared to without downsampling. As a result, Earth+ uses a lower threshold θ to recall those false negatives.

To evaluate the effect of reference image compression, we compress the reference images using different compression ratios, and lower θ properly to align the amount of changed tile between different compression ratios (so that the amount of data need to be downloaded is aligned). In Figure 7, we show that we can compress the reference image by 2600× while only missing 1.7% changed tiles.

Incrementally updating reference images: As Earth+ applies reference-based encoding, which does not encode the unchanged areas in the captured satellite imagery, this saves the on-board storage space used for storing captured imagery by about 80% (since 80% of the areas do not need to be encoded on average, as shown in §6) and enables Earth+ to use the following optimization to further reduce the usage of uplink. Concretely, Earth+ locally *caches* the reference images onboard the satellite for all locations the satellite will visit and only uploads *changed areas* when uploading a new reference image to the satellite. The overhead of such caching is marginal (about 5% compared to the existing storage space used to store observed satellite imagery[7]), and thus fits into the storage space conserved by reference-based encoding. Also, caching reference images on-board allows Earth+ to handle occasional uplink disconnection (more details in §5).

[7]We use DOVEs constellation to estimate this overhead. Let the area that the satellite can download during one ground contact be a km^2. Thus, the storage required to store this a km^2 imagery is $0.87a$ MB, where the coefficient 0.87 is the megabytes required to encode 1km^2 area [26, 35]. Thus, the storage space used to store captured imagery is approximately $2 \times 0.87a = 1.74a$ MB, where this 2× factor is because the ground keeps the captured imagery for two consecutive ground contacts to make sure the downloading is successful [20]. Earth+ stores the reference images of all locations that each satellite will cover, totaling at most $240a$ km^2, where 240 is the current maximum number of ground contact a satellite can have before the satellite finish scanning the whole earth [36]. Since Earth+'s downsampling technique compresses the reference images by 2601×, the total storage space for reference images is at most $240 \times 0.87a$ MB, divided by the compression ratio of Earth+. So the overhead of storing reference is only $0.08a$ MB, 5% of the space for storing captured imagery ($1.74a$ MB).

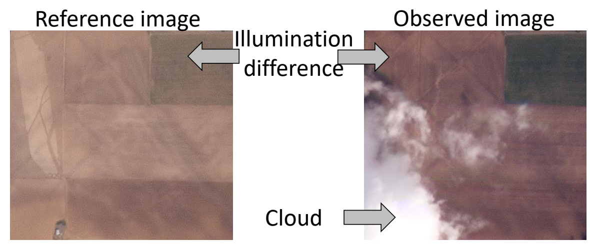


Figure 8. *Two satellite images captured consecutively do have similar content on the ground, but the pixel difference can still be significant due to cloud and illumination differences. Image © 2023 Planet Labs PBC.*

Uploading only cloud-free images: Earth+ requires *cloud-free* reference images to detect terrestrial changes. However, accurately identifying cloud-free imagery on-board can be computationally expensive as it requires neural networks to accurately detect faint clouds [94] and thus not doable on-board. Earth+ thus uses the ground to check if the images are cloud-free retrospectively before uploading it to the satellites.

5 Implementation

Illumination and cloud: In satellite imagery, the time gap between two consecutively-captured images can be hours [74] or days [51]. As a result, two consecutive images in the image sequence can differ a lot in terms of pixel values due to different illumination condition and cloud condition (as illustrated in Figure 8), making the general-purpose change detector (*e.g.,* [45, 63, 81, 89]) no longer suitable for satellite imagery compression.

Note that there are other potential sources (*e.g.* sensor noise, image misalignment) that can also trigger large pixel differences. Earth+ does not explicitly address them, as they only appear in raw data sensed by the satellite, which is not accessible in public datasets.

Filtering out the cloud: Previous work [48, 78] observes that a wide range of applications (*e.g.,* autonomous road detection, precision agriculture) focus on the geographical content on the ground, allowing cloudy areas to be filtered out without impacting analytic results. Based on this observation, Earth+ runs an on-board cloud detector to identify and filter out clouds. However, as accurate cloud detector is too computationally expensive for on-board use (§4.3), Earth+ runs a lightweight decision-tree-based cloud detector instead, which is a widely used cheap cloud detection algorithm that can still detect and filter out most clouds except for faint clouds and haze [53, 79, 90]. These faint cloud and haze are downloaded to the ground by Earth+ (thus increasing the downlink usage of Earth+) but do not affect the analytic results of applications, as these applications will perform accurate cloud removal as an initial pre-processing step .

On-board change detector: Based on the aforementioned cloud filtering mechanism, Earth+ then uses the following workflow to detect changes. First, Earth+ filters out cloud by detecting highly cloudy areas in the satellite imagery using a decision tree classifier, and remove this part of the data. Second, Earth+ drops those images if more than 50% of the areas are filtered by the cloud filter. Third, Earth+ aligns the illumination between the reference image and the captured image on less-cloudy areas using standard linear regression (since the illumination condition affects the pixel value linearly [92]). At last, Earth+ detects, encodes and downloads changes (details in §4.3).

Encoding changed tiles: Earth+ encodes those changed tiles by selecting the changed tiles as region-of-interest and runs region-of-interest encoding on the whole image using an off-the-shelf JPEG-2000 encoder (Kakadu [19]). While encoding such images, Earth+ makes sure that the bit spent on each encoded tile is a constant γ by configuring the bit-per-pixel parameter of the Kakadu encoder as γ times the percentage of tiles that are changed.

Choosing parameters for Earth+: Earth+ introduces two parameters: change detection threshold θ (§4.3) and bit-per-pixel γ. Earth+ chooses θ by profiling last year's data on one single location, and uses this parameter on this year's data for all locations. Earth+ then varies γ to trade-off between downlink usage and imagery quality.

Handling different bands: Unlike traditional RGB images, satellite imagery typically has multiple bands and the amount of changes of different bands are different. For example, vegetation bands measure the concentration of chlorophyll (which is sensitive to temperature), while traditional RGB bands are less sensitive to temperature. To handle such heterogeneity between bands, Earth+ treats each band separately, which means that Earth+ detects changes band-by-band and updates the reference images band-by-band, allowing Earth+ to mark different areas as changed and download different amounts of changes for different bands.

Handling bandwidth fluctuation: To handle uplink fluctuation, as Earth+ locally caches the reference images, Earth+ can randomly skip the updating of some reference images, and instead rely on the cached old reference images (at the cost of downloading more areas). To handle downlink fluctuation, Earth+ leverages the *layered codec* feature, which allows Earth+ to download less layers when downlink is limited (at the cost of degraded the image quality). The feature of layered codec is widely supported by existing imagery encoders on the satellite (*e.g.,* JPEG-2000 encoders [9, 19]).

Updating reference images: Earth+ needs to constantly update its reference images. A naive design is to constantly patch the reference image with newly observed changes, similar to how a video encoder updates its reference images [81, 89]. However, we found that this approach will gradually degrade reference image quality, since each patch will introduce some artifact to the reference image (which is caused by Earth+'s imperfect illumination alignment due to low reference image resolution) and such artifact accumulates.

As a result, Earth+ instead acquires reference images by *whole image downloading*: for each location, Earth+ downloads the first cloud-free image observed by any satellite in the constellation. After this, Earth+ will stop whole image downloading for this location for a month. This operation will not introduce large overhead in large-scale LEO constellations, as the overhead of guaranteed downloading is fixed (at most 12 times a year) and will be evenly spread out by all satellites in the constellation.

Note that although Earth+ starts to find a new cloud-free image as reference one month after observing the last reference image, the extra time it takes to actually find such cloud-free image can be excessively long, which indicates that the actual downloading frequency can be much lower than once per month. In the extreme case, assuming that the constellation contains only *one* satellite, the frequency of whole image downloading will be once every 81 days in average — only around 4 times a year, where this 80-day estimate comes from the fact that Earth+ starts updating the reference after one-month wait, together with an additional wait time of 51 days in average (§3) to actually observe such cloud-free reference image.

6 Evaluation

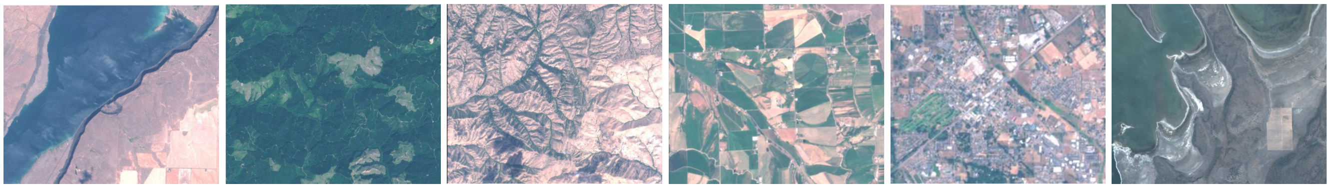
In this section, we pick two state-of-the-art satellite imagery compression systems as our baseline and evaluate Earth+ against on two satellite imagery datasets. The key takeaway of our evaluation is three-fold:

- Compared to the state-of-the-art onboard compression schemes, Earth+ reduces the downlink bandwidth usage by 1.3-3.3× without hurting the imagery quality on not only RGB bands but also other satellite imagery bands.
- These improvements are achieved without using more uplink bandwidth than currently available, or more compute or storage resources than the baselines.
- With more satellites in a constellation, Earth+ can further reduce the amount of downlink bandwidth usage.

6.1 Experimental setup

Dataset: We evaluate Earth+ on two datasets (Table 2 illustrates the details of these two datasets).

Rich-content dataset: We collect 1-year images on 11 geographical locations in Washington State (where each location is of size 1600 km^2) from Sentinel-2 dataset [51]. We sample images from Washington State as it contains a wide variety of geographical contexts, including fluvial landscapes, agricultural areas with varied irrigation systems, mountainous regions with large elevation changes, etc, as shown in Figure 9a-e. Since the total file size of this data is 3.6 TB, to



(a) Sentinel-2: River **(b)** Forest **(c)** Mountain **(d)** Agriculture **(e)** City **(f)** Planet: Coastal

Figure 9. *Snapshots of images at sampled locations in our testing datasets, which show a variety of geographical content.*

Section	Properties	Values
Connectivity	Ground contact duration	*10 minutes* [20, 44]
	Ground contact per day	*7 times* [44]
	Uplink bandwidth	250 kbps [50]
	Downlink bandwidth	200 Mbps [50]
Hardware	On-board storage	*360 GB* [27]
Image	Image resolution	6600×4400 [54]
	Image channels	RGB + InfraRed [54]
	Raw image file size	*150 MB* [26]
	Ground sampling distance	3.7 meters [54]

Table 1. Characterizing the specifications of Doves constellation (2017-2018). Some of the data are not publically available and we infer them from other sources (we *italicized* those data).

handle the large volumn of this dataset, we downsample the images in this dataset by 4×, width and height, where we confirmed on one location that such downsampling does not affect the savings of Earth+.

However, Sentinel-2 dataset [51] only contains two satellites in its constellation. To further show the potential of Earth+'s constellation-wide change-based encoding, we incorporate another dataset with lower coverage but with more satellites available.

Large-constellation dataset: we use Planet dataset [74] that contains multiple satellites in its constellation to showcase the potential of Earth+'s constellation-wide change-based encoding. Due to the quota limit of the Planet dataset, we only sample images on one randomly sampled location in the U.S. (illustrated in Figure 9f), with cloud coverage smaller than 5%. Our sampled dataset contains 48 satellites in total.

Real-world satellite specifications: see Table 1. In this table, we use data from year 2017 to year 2018 as we found the most public satellite specification data during that time period. As a result, such table may not faithfully reflect the specifications of latest satellites.

Uplink and downlink: We use the uplink and downlink specifications from Doves constellation. Specifically:

- Uplink: we assume that the uplink is of 250 kbps [50] and the connection duration is 10 minutes [20, 44]. Here we assume that the uplink bandwidth is a constant, as the uplink leverages the S-band to communicate [8], which is of low frequency, and thus severe weather conditions do not significantly affect its bandwidth [83].
- Downlink: we assume that the ground contact duration is 10 minutes [20, 44] and calculate the average bandwidth required to download a fixed amount of images.

Imagery encoder: We use the off-the-shelf JPEG-2000 encoder called Kakadu [19], which can run on satellite CPU. We note that JPEG-2000 is a variation of JPEG that supports more imagery bands and bit depths and is widely adopted in LEO satellite constellations [51, 74].

Metrics: Earth+ aims to reduce the downlink demand without hurting the quality of downloaded images. We measure the required downlink bandwidth by dividing the amount of downloaded data during one ground contact by the ground contact time (10 minutes [20, 44]) and measure the image quality via Peak Signal-to-Noise Ratio (PSNR for short). This aligns with satellite compression literature [52, 55, 57, 80], and prior work shows that a higher PSNR typically leads to higher application-side performance [43, 62].

We also evaluate the accuracy of vegetation area segmentation on forest areas on one forest location in Sentinel dataset (other locations has low vegetation coverage). We segment the vegetation area by calculating NDVI index [17, 18, 21–24] and thresholding the NDVI index by 0.1 [17]. The accuracy is defined as the percentage of pixels that are correctly identified as vegetation area or non-vegetation area.

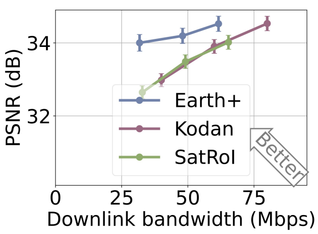
Baselines: We consider two state-of-the-art baselines for on-board satellite imagery compression:

- **Kodan** [48]: drop low-value cloud data and download remaining non-cloudy areas.
- **SatRoI** [78]: run reference-based encoding using the first image for each location in our dataset as the reference image (we make sure that the first image for each location is cloud-free in our dataset).
- **Lossless compression**: compress the satellite imagery using lossless compression. We use two codecs that are commonly used on-board: JPEG2K codec (through Kakadu encoder [19]) and CCSDS 122.0-B-1 codec (via TER encoder [34]).

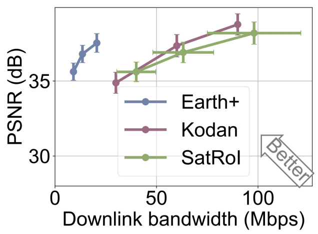
The reason that the SatRoI baseline does not update its reference images via uplink is two-fold. First, the current uplink capacity is insufficient to upload even one reference image (even after the default image compression) for each

	Why using this dataset	Number of satellites included in our dataset	Locations	Coverage of each location	Ground sampling distance (GSD)	Duration	Number of bands	Cloud coverage
Planet	Show that Earth+ saves more downlink when there are more satellites	48	1	36 km^2	3.0 - 4.1 m	3 months	4	<5%
Sentinel-2	Test Earth+ on a wide range of content	2	11	1600 km^2	10 m	1 year	13	≤100%

Table 2. *The datasets used in our evaluation. One from Planet [51] and the other from Sentinel-2 [74].*



(a) Sentinel-2 dataset



(b) Planet dataset

Figure 10. *Earth+ requires less downlink bandwidth without sacrificing the image quality (measured in PSNR). The vertical and horizontal bar shows the standard deviation of the mean (some of the horizontal bars are occluded as they are too short).*

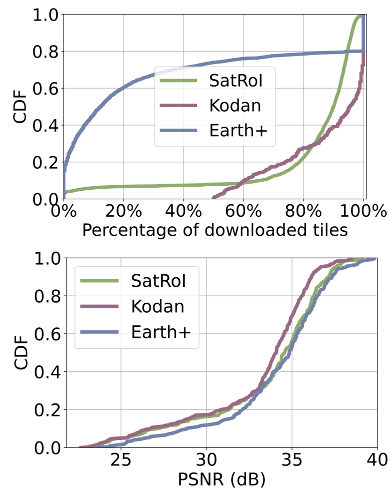


Figure 11. *The CDF of the percentage of downloaded tiles and the PSNR of Earth+ and baselines on Sentinel-2 dataset. We can see that Earth+ downloads much fewer tiles compared to the baselines while achieving higher PSNR.*

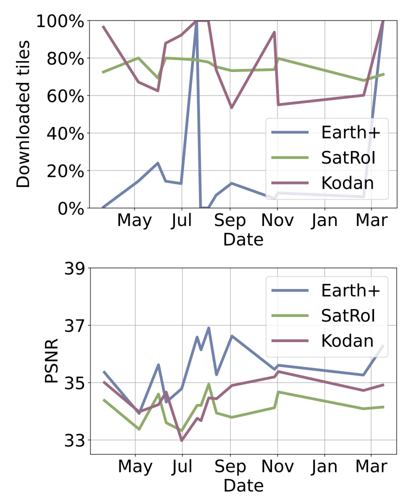


Figure 12. *The time series of the percentage of downloaded tiles and the PSNR of Earth+ and baselines on one location. Earth+ downloads less percentage of tiles than the baselines, unless it performs guaranteed downloading (as described in §5).*

image that will be downloaded during the one-year period of our evaluation. Also, if choosing references from satellite-observed images, SatRoI may frequently use *cloudy* images as references (since 2/3 of the earth is covered by clouds [12] and the on-board cloud detector may incorrectly identify cloudy images as cloud-free, as illustrated in §5), whereas we ensure that the reference images in SatRoI are always cloud-free. As a result, our SatRoI baseline performs strictly better than SatRoI that naively updates reference images.

Also, we evaluate Earth+ using the standard JPEG-2000 image encoder, commonly used by existing satellites [5, 28]. While better satellite imagery encoders exist [37, 40, 67, 75], Earth+ complements these works, as these works focus on how to download the imagery in a target area using less bits, and Earth+ focuses on adjusting the target areas so that those unchanged areas are not downloaded to the ground. We also use JPEG-2000 encoder for other baselines.

6.2 Experimental results

Saving downlink bandwidth without hurting PSNR: As shown in Figure 10a, Earth+ saves 1.3-2.0× downlink bandwidth without hurting the PSNR of the images compared to the strongest baseline in the Sentinel-2 dataset. Earth+ further saves downlink bandwidth by 2.8-3.3× in the Planet dataset, as shown in Figure 10b.

We note that while Earth+ only bridges the huge downlink gap by upto 3.3x, this improvement is still useful as it allows satellites to send imagery to the ground 3.3x more frequently. This does not help applications that can be fully processed in space (*e.g.*, wildfire detection), but it does help other applications (*e.g.*, precision agriculture that also needs farmland sensor data) to make decisions more frequently.

Methods	Downlink Usage	PSNR	Vegetation Accuracy (%)
Earth+	29.84 Mbps	34.73	91.0
SatRoI	40.25 Mbps	33.55	89.9
Kodan	39.68 Mbps	33.04	89.5
JPEG2K	27.40 **Gbps**	∞	100.0
CCSDS	27.64 **Gbps**	∞	100.0

Table 3. Comparing Earth+ against lossy compression approaches (SatRoI, Kodan) and lossless compression schemes (JPEG2K and CCSDS). Earth+ achieves highest accuracy and lowest downlink usage against lossy compression methods, while reducing 1298× downlink usage compared to lossless compression methods with little accuracy loss.

Also, Earth+ achieves a PSNR at around 34-36. Such PSNR is much higher than existing satellite imagery compression work that typically achieves PSNR 30 [52, 55, 57, 80], and is high enough for satellite imagery applications to produce accurate analytic results [43, 62].

Source of improvement: The source of improvement is that Earth+ leverages fresh reference so that fewer areas are changed compared to the reference and downloaded. Concretely, as shown in Figure 11, for more than 60% of the images, Earth+ downloads less than 20% tiles. In contrast, the baseline needs to download more than 80% of tiles for over 70% of images. This is because Earth+ can effectively detect and download only those changed tiles, while Kodan may download those cloud-free but unchanged tiles, and SatRoI cannot effectively detect changes due to using fixed reference images. Note that there are 20% of images fully downloaded by Earth+ because Earth+ guarantees full imagery downloading periodically (§5). Figure 11 also shows that the optimization of Earth+ does not reduce the PSNR.

Time series analysis: Figure 12 shows an illustrative time-series of the downloading behavior of Earth+ and the baselines for one year. Earth+ downloads 5-10× fewer areas than the baselines most of the time. Note that Earth+ downloads 100% area two times a year (in July and March) in Figure 12 due to whole image downloading (§5). Earth+ only performs whole image downloading two times a year, as Earth+ only downloads cloud-free imagery in its entirety but this location is not fully cloud-free most of the time (which can also be validated by that Kodan only captures 4 cloud-free images in a year and downloads 100% area).

Comparing against lossless compression: We further evaluate Earth+ against both lossy compression approaches (SatRoI, Kodan) and lossless compression schemes (JPEG2K and CCSDS) in Table 3. In this set of experiments, we pick the forest location from Sentinel dataset (other locations has low vegetation coverage) and analyze the PSNR and vegetation area segmentation accuracy of that location. Earth+ improves the accuracy by 1.1% and lower the downlink usage by 1.3× against lossy compression approaches like Kodan and SatRoI, while cutting down the downlink usage by 1298× compared to lossless compression schemes like JPEG2K and CCSDS while still maintaining an accuracy greater than 90%.

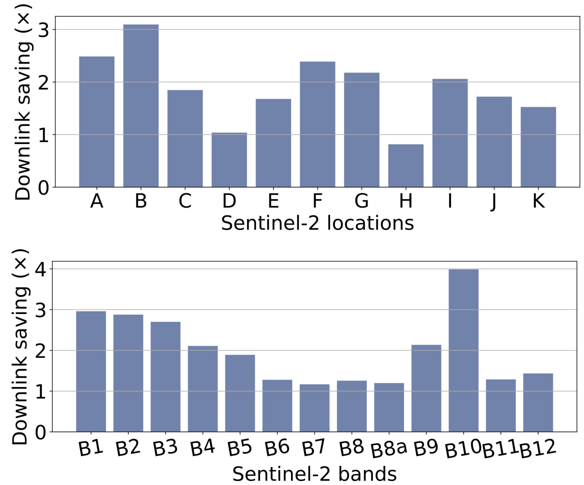


Figure 13. *The improvement of Earth+ across different locations and satellite imagery bands in Sentinel-2 dataset.*

Downlink saving across different locations: We then calculated Earth+'s saving on the downlink (defined as the strongest baseline with lower PSNR than Earth+, divided by the downlink usage of Earth+) grouped by 11 different locations in Washington State. Figure 13 shows that Earth+ is better than the strongest baseline at 10 out of 11 locations. However, Earth+ does not improve on H and has only marginal improvement on D, as these two locations are highly snowy during winter and spring, and snow albedo (*i.e.,* the reflectance of snow) is constantly changing (*e.g.,* old snow has a lower albedo than fresh snow, and dirty snow has a lower albedo than clean snow). Thus, Earth+ tends always to detect changes and download those tiles that contain snow, lowering the improvement of Earth+.

Downlink saving across different bands: We further evaluate the downlink saving of Earth+ (*i.e.,* the downlink usage of the strongest baseline with lower PSNR than Earth+, divided by the downlink usage of Earth+) across different satellite imagery bands. As shown in Figure 13, Earth+ can improve on all 13 bands in Sentinel-2 images, with more improvements on slowly-changing bands (*e.g.,* traditional RGB bands B2-4) but less on fast-changing bands (*e.g.,* vegetation bands B7-8a, which change a lot due to the sensitivity to the temperature).

Storage overhead: Based on Dove [25] constellation specification, we calculate the storage requirements for Earth+ and two baselines. As shown in Figure 14, the onboard storage requirements are 30 GB, 255 GB, and 24 GB for SatRoI, Kodan, and Earth+, respectively. We note that Earth+ spends less storage space to store the captured imagery, as Earth+ only stores the *changed tiles* on-board. Further, Earth+ needs much

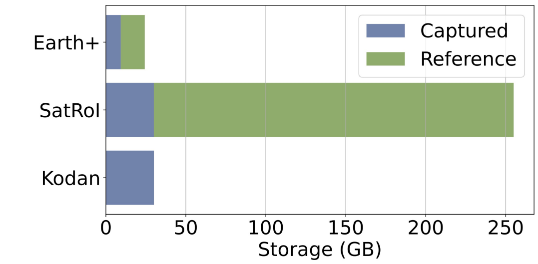


Figure 14. *Breaking down the storage usage of Earth+ and the baselines. Earth+ saves changed areas on-board rather than whole images like Kodan, which leaves some space for storing reference images.*

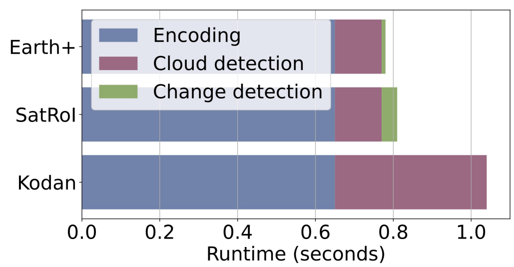


Figure 15. *Benchmarking the runtime of Earth+ and baselines to process one single satellite imagery. The runtime of Earth+ is lower than the baselines.*

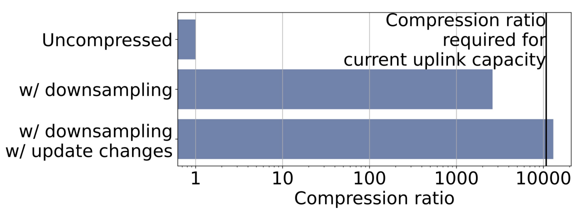


Figure 16. *Earth+ compresses the reference image by over 10,000× so that they can fit in the limited uplink capacity, indicated by the black vertical line.*

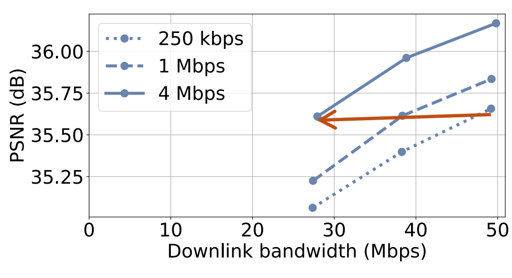


Figure 17. *The PSNR – downlink bandwidth trade-off of Earth+ under different uplink bandwidth. Earth+ can further reduce the downlink bandwidth usage by 22 Mbps by increasing the uplink bandwidth to 4 Mbps (as indicated by the red arrow).*

less storage space for reference images, as Earth+ downsamples the reference images to low resolution. This is because Earth+ stores only changed areas on-board, thus squeezing out storage space to store reference images.

Uplink overhead: We then evaluate the effectiveness of techniques proposed in §4.3 that reduce the update bandwidth usage based on the Sentinel-2 dataset and Dove specification. Figure 16 shows that after applying reference image downsampling and uploading the changed tiles in reference images, Earth+ compresses the reference image by over 10000× and meets the requirement of the limited uplink bandwidth.

Computation overhead: We also measure the runtime that Earth+ and the baselines take to process one imagery, using AMD EPYC 7452 CPU with 16 cores. As shown in Figure 15, the runtime of Earth+ is the lowest. Concretely, both Earth+ and the baselines take 0.65 seconds to encode the imagery. However, Kodan uses an expensive cloud detector that takes 0.39 seconds to run, while Earth+ and SatRoI use cheap cloud detector that only takes 0.12 seconds to run. Further, Earth+ uses *downsampled* reference images to detect changes, allowing the change detection process to run faster.

We note that this evaluation is performed on a server-level CPU so the latency numbers do not reflect the real latency when running Earth+ in a real satellite. That said, our results are indicative to show that Earth+ has lower computation overhead (and thus lower energy consumption).

More uplink, more improvement: The performance of Earth+ (in terms of PSNR – downlink bandwidth trade-off) can be further improved with more uplink capacity, as shown in Figure 17. We highlight that Earth+ can further reduce the downlink bandwidth usage by 22 Mbps by increasing the uplink to 4 Mbps.

More satellites, more improvement: To verify that Earth+ can achieve higher compression by using reference images generated from the whole constellation, we conduct a simulation based on the Planet thumbnail images (we use thumbnail images to bypass the quota limit of the Planet dataset) collected in Denver from July 1st, 2023 to October 1st, 2023. We normalize the illumination condition of these images and run change detection algorithm on top of them. We then calculate the compression ratio based on the average changed areas (*e.g.*, 10% changed areas on average means that Earth+ needs to download 10% areas each time, which corresponds to 10× compression)[8]. Our experimental results show that Earth+ can only achieve 3× compression in a one-satellite constellation. In contrast, Earth+ can achieve 7.3× compression in a 16-satellite constellation.

[8]We point out that this is a rough estimation, as in practice image codec's compression efficiency may decrease when encoding a small amount of areas.

7 Related Work

Single-image compression: A wide range of prior work has focused on single-image compression by augmenting traditional image codecs like JPEG-2000 [37–39, 58, 67, 75, 76, 93] or developing neural-based codec such as autoencoders [40, 41, 47, 95, 96]. Earth+ complements these works, as these works focus on how to download the imagery in a target area using less bits, and Earth+ focuses on adjusting the target areas so that those unchanged areas are not downloaded to the ground.

Change-based encoding: A rich set of literature aims to further compress images by detecting changes *between* images. A line of work builds video-based codecs (*e.g.,* H.264 [89], H.265 [81], VP8 [42], VP9 [71] and autoencoders [64, 65, 91]) to leverage such redundancy, with the assumption that two consecutive captures have similar pixel values. This is not true for satellite imagery due to varying cloud and illumination conditions. Another line of work [78, 88] develops change-based encoding that is robust to varying cloud and illumination conditions. Earth+ also falls into this category. However, existing work can only update the reference image using single-satellite information, while Earth+ allows updating the reference image using images from *all* the satellites in the same constellation, resulting in a fresher reference image and, thus, better change-based encoding quality.

In-orbit computing: An alternative way to reduce the total downlink capacity is to have a concrete application in mind and drop out images that are irrelevant to this application (*e.g.,* [48, 49, 82]). However, this approach may drop out images that are crucial for other applications. In contrast, Earth+ only drops areas that are unchanged, allowing Earth+ to be used by a wider range of applications.

Multipath imagery delivering: One can reduce the latency of obtaining newly-observed satellite imagery by enabling multiple satellites to download the same imagery using inter-satellite links [61, 66]. However, this approach does not increase the amount of imagery downloaded to the ground, as it does not increase the total downlink capacity of the constellation, or reduce the total amount of imagery data need to be downloaded. In contrast, Earth+ allows more imagery to be downloaded to the ground.

8 Limitation

While Earth+ improves satellite imagery compression, several concerns remain.

Lossy compression: Earth+'s compression is lossy. While it allows downloading more images, lossy compression may not be applicable to applications that require lossless compression. To address this issue, future work can improve the image quality of Earth+ by augmenting uplink bandwidth. Also, one can repurpose Earth+ for lossless compression by performing lossless delta-based compression.

Evaluating on ground-processed imagery: Due to the lack of raw satellite imagery data (*i.e.,* Level-0 imagery data) in public datasets, we evaluate Earth+ on public imagery that is post-processed by the ground, which did not faithfully reflect the impact of geographical misalignment and sensor noise on Earth+. That said, we believe this issue is not severe as Earth+ detects changes using low-resolution reference images, which is less sensitive to misalignment and noises compared to full-resolution reference images.

Control messages: Earth+ uses the uplink bandwidth that is reserved for control messages to upload reference images. That said, we believe this is not a serious practical concern as the bandwidth needed for ground-to-satellite control messages is low (*e.g.,* 2.4 kbps [70]) and do not currently use much of the uplink bandwidth capacity (*e.g.,* 250 kbps [50]).

Generalization of results: Our evaluation of Earth+ focuses on a specific set of satellite specs and imagery datasets, but it does not show how effective Earth+ would be if it is used on other or future earth-observation satellites. We hope our work will inspire more research to examine Earth+ in other environments.

Deployment concerns: Though Earth+ only changes software, there may be complications in implementing Earth+ on existing satellites as Earth+ requires a software update on the satellite's imagery encoding module onboard the satellite.

Stepping back, we acknowledge that Earth+ does increase the system complexity, especially on the ground stations, including sharing downloaded images across ground stations efficiently. However, we believe Earth+ takes the first step towards delivering more images to the ground by constellation-wide imagery sharing.

9 Conclusion

While satellite imagery is useful for a wide range of applications, most of the imagery observed by the satellites is not downloaded to the ground due to limited downlink capacity. This work presents Earth+, a new onboard satellite imagery compression system to reduce the downlink bandwidth usage. Earth+ is the first to leverage images across an entire satellite constellation to allow downloading more images to the ground. Earth+ further uses several techniques to judiciously select and upload reference images under limited uplink capacity. We show that Earth+ can compress the imagery by upto 3.3× without compromising imagery quality on all bands or using more computation and storage resources, while staying within real-world uplink constraints.

Acknowledgments

We thank the anonymous reviewers and our shepherd Nathaniel Bleier. The project is funded by NSF CNS-2146496, CNS-2131826, CNS-2313190, CNS-1901466, CNS-2313190, and research awards from Google, CERES Center, Conviva, and Chameleon Cloud.

References

[1] Aqua - part of the a-train satellites | nasa. https://www.nasa.gov/mission_pages/aqua/index.html. (Accessed on 09/11/2023).

[2] Axiom space collaborates to advance in-space data processing and cybersecurity solutions — axiom space. https://www.axiomspace.com/news/snowcone. (Accessed on 10/22/2024).

[3] Biomass (satellite) - wikipedia. https://en.wikipedia.org/wiki/Biomass_(satellite). (Accessed on 08/20/2023).

[4] Celestrak: Special data request form. https://celestrak.org/NORAD/archives/request.php. (Accessed on 01/15/2024).

[5] Data formats - user guides - sentinel-2 msi - sentinel online - sentinel online. https://sentinels.copernicus.eu/web/sentinel/user-guides/sentinel-2-msi/data-formats. (Accessed on 02/02/2024).

[6] Environmental management, environmental monitoring | satellite imaging corp. https://www.satimagingcorp.com/applications/environmental-impact-studies/. (Accessed on 09/08/2023).

[7] Esa - low earth orbit. https://www.esa.int/ESA_Multimedia/Images/2020/03/Low_Earth_orbit. (Accessed on 12/04/2023).

[8] Esa - sentinel-2 operations. https://www.esa.int/Enabling_Support/Operations/Sentinel-2_operations. (Accessed on 12/04/2023).

[9] Github - uclouvain/openjpeg: Official repository of the openjpeg project. https://github.com/uclouvain/openjpeg. (Accessed on 01/16/2024).

[10] Global forest area 2020 | statista. https://www.statista.com/statistics/1292175/global-forest-area/. (Accessed on 08/20/2023).

[11] How much do satellites cost? - how satellites work | howstuffworks. https://science.howstuffworks.com/satellite10.htm#:~:text=A%20typical%20weather%20satellite%20carries,of%20maintaining%20and%20repairing%20satellites. (Accessed on 12/03/2023).

[12] How much of the earth is covered by clouds? - geography realm. https://www.geographyrealm.com/how-much-of-the-earth-is-covered-by-clouds/. (Accessed on 11/16/2023).

[13] How much of the ocean has been explored? : Ocean exploration facts: Noaa office of ocean exploration and research. https://oceanexplorer.noaa.gov/facts/explored.html. (Accessed on 08/21/2023).

[14] How satellite imagery is helping precision agriculture grow to new heights | news | eco-business | asia pacific. https://www.eco-business.com/news/how-satellite-imagery-is-helping-precision-agriculture-grow-to-new-heights/. (Accessed on 12/03/2023).

[15] In-space production applications - iss national lab. https://issnationallab.org/research-and-science/space-research-overview/research-areas/in-space-production-applications/. (Accessed on 10/22/2024).

[16] in-space-production-applications-overview-october-2023.pdf. https://www.nasa.gov/wp-content/uploads/2023/10/in-space-production-applications-overview-october-2023.pdf. (Accessed on 10/22/2024).

[17] ipad.fas.usda.gov/cropexplorer/definitions/spotveg.htm. https://ipad.fas.usda.gov/cropexplorer/Definitions/spotveg.htm. (Accessed on 11/02/2024).

[18] ipad.fas.usda.gov/cropexplorer/definitions/spotveg.htm. https://ipad.fas.usda.gov/cropexplorer/Definitions/spotveg.htm. (Accessed on 11/03/2024).

[19] Kakadu software. https://kakadusoftware.com/. (Accessed on 09/05/2023).

[20] leoconn_2021_day1_kiruthika.pdf. https://leoconn.github.io/slides/leoconn_2021_day1_kiruthika.pdf. (Accessed on 09/10/2023).

[21] Measuring vegetation (ndvi & evi). https://earthobservatory.nasa.gov/features/MeasuringVegetation/measuring_vegetation_2.php. (Accessed on 11/03/2024).

[22] Ndvi mapping in agriculture, index formula, and uses. https://eos.com/make-an-analysis/ndvi/. (Accessed on 11/03/2024).

[23] Normalized difference vegetation index - an overview | sciencedirect topics. https://www.sciencedirect.com/topics/earth-and-planetary-sciences/normalized-difference-vegetation-index#:~:text=Normalized%20difference%20vegetation%20index%20(NDVI)%20is%20a%20simple%20graphical%20indicator,green%20health%20vegetation%20or%20not. (Accessed on 11/03/2024).

[24] Normalized difference vegetation index - wikipedia. https://en.wikipedia.org/wiki/Normalized_difference_vegetation_index. (Accessed on 11/03/2024).

[25] Planet | insights - our constellations. https://www.planet.com/our-constellations/. (Accessed on 08/30/2023).

[26] Planet_fusion_specification_v1.0.0. https://assets.planet.com/docs/Fusion-Tech-Spec_v1.0.0.pdf. (Accessed on 11/09/2023).

[27] Planetscope - high-resolution satellite imagery source | pacgeo.com. https://pacgeo.com/satellite/planetscope/. (Accessed on 02/02/2024).

[28] Planetscope2 imagery available in the planet platform. https://www.planet.com/pulse/planetscope2-imagery-available-in-the-planet-platform/. (Accessed on 02/02/2024).

[29] Precision agriculture imaging with planet satellite solutions | planet. https://www.planet.com/markets/monitoring-for-precision-agriculture/https://www.planet.com/markets/monitoring-for-precision-agriculture/. (Accessed on 09/08/2023).

[30] Precision agriculture imaging with planet satellite solutions | planet. https://www.planet.com/markets/monitoring-for-precision-agriculture/. (Accessed on 12/03/2023).

[31] Satellite imagery road segmentation | by nithish | medium. https://medium.com/@nithishmailme/satellite-imagery-road-segmentation-ad2964dc3812. (Accessed on 09/08/2023).

[32] Satellite images, satellite map | satellite imaging corp. https://www.satimagingcorp.com/gallery/high-resolution/. (Accessed on 12/06/2023).

[33] Sentinel 2 bands and combinations - gis geography. https://gisgeography.com/sentinel-2-bands-combinations/. (Accessed on 12/06/2023).

[34] Ter software. http://www.gici.uab.cat/TER/. (Accessed on 11/02/2024).

[35] Understanding planetscope instruments. https://developers.planet.com/docs/apis/data/sensors/. (Accessed on 09/11/2023).

[36] User guides - sentinel-2 msi - resolutions - sentinel online - sentinel online. https://sentinels.copernicus.eu/web/sentinel/user-guides/sentinel-2-msi/resolutions. (Accessed on 09/09/2023).

[37] Masud Ibn Afjal, Md Al Mamun, and Md Palash Uddin. Band reordering heuristics for lossless satellite image compression with 3d-calic and ccsds. *Journal of Visual Communication and Image Representation*, 59:514–526, 2019.

[38] Masud Ibn Afjal, Palash Uddin, Al Mamun, and Abu Marjan. An efficient lossless compression technique for remote sensing images using segmentation based band reordering heuristics. *International Journal of Remote Sensing*, 42(2):756–781, 2021.

[39] Bruno Aiazzi, Luciano Alparone, Stefano Baronti, and Cinzia Lastri. Crisp and fuzzy adaptive spectral predictions for lossless and near-lossless compression of hyperspectral imagery. *IEEE Geoscience and Remote Sensing Letters*, 4(4):532–536, 2007.

[40] Vinicius Alves de Oliveira, Marie Chabert, Thomas Oberlin, Charly Poulliat, Mickael Bruno, Christophe Latry, Mikael Carlavan, Simon Henrot, Frederic Falzon, and Roberto Camarero. Reduced-complexity end-to-end variational autoencoder for on board satellite image compression. *Remote Sensing*, 13(3):447, 2021.

[41] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436*, 2018.

[42] Jim Bankoski, Paul Wilkins, and Yaowu Xu. Technical overview of VP8, an open source video codec for the web. In *2011 IEEE International Conference on Multimedia and Expo*, pages 1–6. IEEE, 2011.

[43] Marco Cagnazzo, Sara Parrilli, Giovanni Poggi, and Luisa Verdoliva. Costs and advantages of object-based image coding with shape-adaptive wavelet transform. *EURASIP Journal on Image and Video Processing*, 2007:1–13, 2007.

[44] Shkelzen Cakaj, Bexhet Kamo, Algenti Lala, and Alban Rakipi. The coverage analysis for low earth orbiting satellites at low elevation. *International Journal of Advanced Computer Science and Applications*, 5(6), 2014.

[45] Tiffany Yu-Han Chen, Lenin Ravindranath, Shuo Deng, Paramvir Bahl, and Hari Balakrishnan. Glimpse: Continuous, real-time object recognition on mobile devices. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, pages 155–168, 2015.

[46] Yang Cunjian, Wang Siyuan, Zhang Zengxiang, Huang Shifeng, et al. Extracting the flood extent from satellite SAR image with the support of topographic data. In *2001 International Conferences on Info-Tech and Info-Net. Proceedings (Cat. No. 01EX479)*, volume 1, pages 87–92. Ieee, 2001.

[47] Vinicius Alves de Oliveira, Marie Chabert, Thomas Oberlin, Charly Poulliat, Mickael Bruno, Christophe Latry, Mikael Carlavan, Simon Henrot, Frederic Falzon, and Roberto Camarero. Satellite image compression and denoising with neural networks. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2022.

[48] Bradley Denby, Krishna Chintalapudi, Ranveer Chandra, Brandon Lucia, and Shadi Noghabi. Kodan: Addressing the computational bottleneck in space. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3*, pages 392–403, 2023.

[49] Bradley Denby and Brandon Lucia. Orbital edge computing: Nanosatellite constellations as a new class of computer system. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 939–954, 2020.

[50] Kiruthika Devaraj, Ryan Kingsbury, Matt Ligon, Joseph Breu, Vivek Vittaldev, Bryan Klofas, Patrick Yeon, and Kyle Colton. Dove high speed downlink system. 2017.

[51] Matthias Drusch, Umberto Del Bello, Sébastien Carlier, Olivier Colin, Veronica Fernandez, Ferran Gascon, Bianca Hoersch, Claudia Isola, Paolo Laberinti, Philippe Martimort, et al. Sentinel-2: Esa's optical high-resolution mission for gmes operational services. *Remote sensing of Environment*, 120:25–36, 2012.

[52] Lilian N Faria, Leila MG Fonseca, and Max HM Costa. Performance evaluation of data compression systems applied to satellite imagery. *Journal of Electrical and Computer Engineering*, 2012:18–18, 2012.

[53] Hualian Fu, Yuan Shen, Jun Liu, Guangjun He, Jinsong Chen, Ping Liu, Jing Qian, and Jun Li. Cloud detection for fy meteorology satellite based on ensemble thresholds and random forests approach. *Remote Sensing*, 11(1):44, 2018.

[54] Sajid Ghuffar. Dem generation from multi satellite planetscope imagery. *Remote Sensing*, 10(9):1462, 2018.

[55] KS Gunasheela and HS Prasantha. Satellite image compression-detailed survey of the algorithms. In *Proceedings of International Conference on Cognition and Recognition: ICCR 2016*, pages 187–198. Springer, 2018.

[56] Tobias B Hank, Katja Berger, Heike Bach, Jan GPW Clevers, Anatoly Gitelson, Pablo Zarco-Tejada, and Wolfram Mauser. Spaceborne imaging spectroscopy for sustainable agriculture: Contributions and challenges. *Surveys in Geophysics*, 40(3):515–551, 2019.

[57] A Indradjad, Ali Syahputra Nasution, Hidayat Gunawan, and Ayom Widipaminto. A comparison of satellite image compression methods in the wavelet domain. In *IOP Conference Series: Earth and Environmental Science*, volume 280, page 012031. IOP Publishing, 2019.

[58] Sushil K Jain and Donald A Adjeroh. Edge-based prediction for lossless compression of hyperspectral images. In *2007 Data Compression Conference (DCC'07)*, pages 153–162. IEEE, 2007.

[59] Kazuro Kikuchi. Fundamentals of coherent optical fiber communications. *Journal of Lightwave Technology*, 34(1):157–179, 2016.

[60] George Kopsiaftis and Konstantinos Karantzalos. Vehicle detection and traffic density monitoring from very high resolution satellite video data. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1881–1884. IEEE, 2015.

[61] Zeqi Lai, Qian Wu, Hewu Li, Mingyang Lv, and Jianping Wu. Orbitcast: Exploiting mega-constellations for low-latency earth observation. In *2021 IEEE 29th International Conference on Network Protocols (ICNP)*, pages 1–12. IEEE, 2021.

[62] F Li, V Lukin, G Proskura, I Vasilyeva, and G Chernova. Image classification accuracy analysis for three-channel remote sensing data. In *Proceedings of the 3rd International Workshop on Intelligent Information Technologies & Systems of Information Security (IntellTSIS 2022), Khmelnytskyi, Ukraine*, pages 505–519, 2022.

[63] Yuanqi Li, Arthi Padmanabhan, Pengzhan Zhao, Yufei Wang, Guoqing Harry Xu, and Ravi Netravali. Reducto: On-camera filtering for resource-efficient real-time video analytics. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*, pages 359–376, 2020.

[64] Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, and Zhiyong Gao. Dvc: An end-to-end deep video compression framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11006–11015, 2019.

[65] Guo Lu, Xiaoyun Zhang, Wanli Ouyang, Li Chen, Zhiyong Gao, and Dong Xu. An end-to-end learning framework for video compression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3292–3308, 2021.

[66] Mingyang Lyu, Qian Wu, Zeqi Lai, Hewu Li, Yuanjie Li, and Jun Liu. Falcon: Towards fast and scalable data delivery for emerging earth observation constellations. In *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*, pages 1–10. IEEE, 2023.

[67] Enrico Magli, Gabriella Olmo, and Emanuele Quacchio. Optimized onboard lossless and near-lossless compression of hyperspectral data using calic. *IEEE Geoscience and remote sensing letters*, 1(1):21–25, 2004.

[68] Salvatore Manfreda, Matthew F McCabe, Pauline E Miller, Richard Lucas, Victor Pajuelo Madrigal, Giorgos Mallinis, Eyal Ben Dor, David Helman, Lyndon Estes, Giuseppe Ciraolo, et al. On the use of unmanned aerial systems for environmental monitoring. *Remote sensing*, 10(4):641, 2018.

[69] Jonathan C McDowell. The low earth orbit satellite population and impacts of the spacex starlink constellation. *The Astrophysical Journal Letters*, 892(2):L36, 2020.

[70] Margaret M McMahon and Robert Rathburn. Measuring latency in iridium satellite constellation data services. *US Naval Academy Report*, (A291464):68, 2005.

[71] Debargha Mukherjee, Jim Bankoski, Adrian Grange, Jingning Han, John Koleszar, Paul Wilkins, Yaowu Xu, and Ronald Bultje. The latest open-source video codec vp9-an overview and preliminary results. In *2013 Picture Coding Symposium (PCS)*, pages 390–393. IEEE, 2013.

[72] David J Mulla. Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps. *Biosystems engineering*, 114(4):358–371, 2013.

[73] Thanh Tam Nguyen, Thanh Dat Hoang, Minh Tam Pham, Tuyet Trinh Vu, Thanh Hung Nguyen, Quyet-Thang Huynh, and Jun Jo. Monitoring agriculture areas with satellite images and deep learning. *Applied Soft Computing*, 95:106565, 2020.

[74] Planet Labs PBC. Planet application program interface: In space for life on earth, 2018–.

[75] Barbara Penna, Tammam Tillo, Enrico Magli, and Gabriella Olmo. Progressive 3-d coding of hyperspectral images based on jpeg 2000. *IEEE Geoscience and remote sensing letters*, 3(1):125–129, 2006.

[76] Francesco Rizzo, Bruno Carpentieri, Giovanni Motta, and James A Storer. Low-complexity lossless compression of hyperspectral imagery via linear prediction. *IEEE Signal Processing Letters*, 12(2):138–141, 2005.

[77] Takashi Sasaki, Morio Toyoshima, and Hideki Takenaka. Digital coherent optical receiver for satellite laser communication. In *2011 International Conference on Space Optical Systems and Applications (ICSOS)*, pages 245–247, 2011.

[78] Christofer Schwartz, Ingo Sander, Fredrik Bruhn, Mathias Persson, Joakim Ekblad, and Christer Fuglesang. Satellite image compression guided by regions of interest. *Sensors*, 23(2):730, 2023.

[79] Smadar Shiffman and Ramakrishna Nemani. Evaluation of decision trees for cloud detection from avhrr data. Technical report, 2005.

[80] Halah Saadoon Shihab, Suhaidi Shafie, Abdul Rahman Ramli, and Fauzan Ahmad. Enhancement of satellite image compression using a hybrid (dwt–dct) algorithm. *Sensing and Imaging*, 18:1–30, 2017.

[81] Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on circuits and systems for video technology*, 22(12):1649–1668, 2012.

[82] Bill Tao, Om Chabra, Ishani Janveja, Indranil Gupta, and Deepak Vasisht. Known knowns and unknowns: Near-realtime earth observation via query bifurcation in serval.

[83] Siva Priya Thiagarajah, Shamini Pillay, Shalini Darmaraju, Rajamani Subramanian, and Michelle Foo May Fung. The effect of rain attenuation on s-band terrestrial links. In *2013 IEEE Symposium on Wireless Technology & Applications (ISWTA)*, pages 192–197. IEEE, 2013.

[84] David M Tralli, Ronald G Blom, Victor Zlotnicki, Andrea Donnellan, and Diane L Evans. Satellite remote sensing of earthquake, volcano, flood, landslide and coastal inundation hazards. *ISPRS Journal of Photogrammetry and Remote Sensing*, 59(4):185–198, 2005.

[85] Compton J Tucker and John RG Townshend. Strategies for monitoring tropical deforestation using satellite data. *International Journal of Remote Sensing*, 21(6-7):1461–1471, 2000.

[86] Adam Van Etten, Dave Lindenbaum, and Todd M Bacastow. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232*, 2018.

[87] Shane M Walsh, Skevos FE Karpathakis, Ayden S McCann, Benjamin P Dix-Matthews, Alex M Frost, David R Gozzard, Charles T Gravestock, and Sascha W Schediwy. Demonstration of 100 gbps coherent free-space optical communications at leo tracking rates. *Scientific Reports*, 12(1):18345, 2022.

[88] Xu Wang, Ruimin Hu, Zhongyuan Wang, and Jing Xiao. Virtual background reference frame based satellite video coding. *IEEE Signal Processing Letters*, 25(10):1445–1449, 2018.

[89] Thomas Wiegand, Gary J Sullivan, Gisle Bjontegaard, and Ajay Luthra. Overview of the h. 264/avc video coding standard. *IEEE Transactions on circuits and systems for video technology*, 13(7):560–576, 2003.

[90] Lina Xu, Shenghui Fang, Ruiqing Niu, and Jiong Li. Cloud detection based on decision tree over tibetan plateau with modis data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 39:535–538, 2012.

[91] Ren Yang, Fabian Mentzer, Luc Van Gool, and Radu Timofte. Learning for video compression with hierarchical quality and recurrent enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6628–6637, 2020.

[92] Xiajun Yang, CP Lo, et al. Relative radiometric normalization performance for change detection from multi-date satellite images. *Photogrammetric engineering and remote sensing*, 66(8):967–980, 2000.

[93] Jing Zhang and Guizhong Liu. An efficient reordering prediction-based lossless compression algorithm for hyperspectral images. *IEEE Geoscience and remote sensing letters*, 4(2):283–287, 2007.

[94] Mingmin Zhao, Peder Olsen, and Ranveer Chandra. Seeing through clouds in satellite images. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[95] Lei Zhou, Chunlei Cai, Yue Gao, Sanbao Su, and Junmin Wu. Variational autoencoder for low bit-rate image compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 2617–2620, 2018.

[96] Lei Zhou, Zhenhong Sun, Xiangji Wu, and Junmin Wu. End-to-end optimized image compression with attention mechanism. In *CVPR workshops*, page 0, 2019.